**Sudan University Of science And Technology**

College of Computer Science And Technology

Proposal for Bachelors degree in Computer Science

Title of the project:

**RESTful API TESTING TOOL**

Supervisor:

Dr. Nahla murtada

Nov/2019

# Overview:

**API testing (Application Programming Interface Testing) is a software testing type which focuses on the determination if the developed APIs meet expectations regarding the functionality, reliability, performance, and security of the application.**

In APIs development testing is always necessary to ensure the API is functioning properly and responding with the correct results in the correct format.

For API testing we would need an application/tool to interact with the API and then would write our code/endpoints to test the API and since interacting with APIs/web services has grown significantly over the last couple of years and is expected to grow 30% more according to a research by Smart bear then API testing tools are needed more than ever.

From all of the above we conclude that API automation testing is now a crucial part of a successful CI/ DevOps practice.

**Motivation**:

RESTful web services are being used widely nowadays. RESTful is simple and most widely used among other web services. In fact, its simplicity is a reason for its fame as well.

1-a lot of clients use rest API to transfer data RESTful API nowadays used in IOT connect the controller with server, web apps, mobile app, etc.

2-reuse and reusability API can be reused as service to handle the specific problems so a reused service must be tested and have well-formed docs to able to use.

3-Team management and collaboration are important things to get work done by managing their work and responsibility to reduce the risk of miscommunication and increase the productivity of team members by using tools and easy ways to track work and manage the work.

4-App/systems have different versions and each version may be developed by different development teams so make each version stick with it API documentation reduces maintenance work.

**Introduction:**

The project is a testing tool that aims to ease the API development with test automation, documentation generation with templates, team collaboration, version control.

The tool will allow you to send RESTful API requests directly and receive and view the response or the error code if there’s an error.

The tester can view the response to confirm that it’s in the right shape, prepare templates for it, write the documentation description for each endpoint.

The tool will also ease the team collaboration by giving a way for the team to communicate with each other also will ease the maintenance by implementing version controlling so the developer can role back to earlier versions if something went wrong.

# Research Problem:

Let’s assume that there’s a project with a developer working on the API and some other developer (could be outside the project) who’s using this API and he discovered a malfunction or a bug with the API in this case the API user will have to inform the API developer of this bug so that the API developer can fix it and then re-inform the developer that the bug has been fixed.

Putting in mind that this process might be repeated several times and also there could be more than one API (possibly hundreds or even thousands of APIs in development). We can see how time and effort consuming this process is.

This is where the API tester comes in hand by testing the API and making sure it’s working properly before sending it back to the user/developer.

First problem here comes in testing the app since RESTful API requests naturally work between clients and servers where the client sends a request to the server and the server responds back with the data, the problem here is that in order for the API tester to do his job the API user will have to connect him with the client app so he’ll be able to do the GUI testing and this is where costs increase and the tester will have to do the testing work on the app GUI.

The tool here will simulate the client by sending requests and showing the response to the tester, and making it even easier by adding test automation with test suits and cases, this way the tester won’t have to use the client app GUI by just using the tool instead.

Second problem here comes in writing the documentations for the API since the API user needs to know the API’s endpoints, what they receive, the responses, the response’s format and so on.

Writing documentation manually includes manually writing what each endpoint receives and returns and their descriptions which is very expensive in terms of time and cost and unnecessary extra work for the tester. And any misspelling or fault could lead to a confusing or ambiguous API for the consumer.

With the API testing automation tool, the tester doesn’t need to worry about writing the documentation the tool will auto generate the documentations once the tests reach a full coverage.

Another problem with documentation is version controlling in CI/CD where we will need a new different version of documentations for each version of the API for the later maintenance.

Third problem is communication between collaborators since they don’t have to be all working under the same company or organization since the API developer have to inform the tester whenever he changes or adds new features to the API and then the tester will need to re-inform him back with the test results and the documentations for the API users thus we need a way of connecting all of the developers and testers in one platform and this is where the team collaboration part of the tool comes in handy.

# Questions:

# There are some important questions will be addressed in this research which are

1. What is the useful of this project?
2. How can we apply this project?
3. Who are the beneficiaries of the project?
4. What are the processing stages?
5. What are the technologies used in the project?
6. What are the techniques used in the processing?

**Impacts:**

Impact whole development team including front end developer and backend developer and the manager.

Make rest API documentation clear for front end developers and easy to connect with clients.

Developer or tester can test his work before delivering it so fewer bugs in productions

Manager can review developer workflow and how he/she develop and API

And also, the manager can monitor front end developers about the connection with the client.

Maintenance team: continuous delivery and docs allow the next team that will use this app to recreate anew rest full API docs.

Value of using this platform to have well-formed restful API documentation that all endpoint within this document are tested and passed the tests through automation testing, and have good data represent each action done by the user, to help in management actions and decisions making

**API testing is the only way to provide truly secure, reliable and scalable connections between platforms. Testing provides these benefits:**

1. Access to the server without user interface
2. Time Efficiency
3. Cost-Effective / Reduces Testing Cost

**Scope:**

user can use our software by create an account in our platform after that he/she can create a project and this project include admin (the creator of the project), developer or tester in our scope we expect developer the person that who test API endpoint because we focus on small teams and we use DevOps method that based on small teams with greats abilities and the consumer of APIs endpoint (front end developer that use APIs in clients platforms).

Admin sends invitations to all stakeholders in their email if they already have an account in our platform after the accept invitations any stakeholder has roles and responsibilities and work to do.

Developer or tester create request by providing request information’s and add title and descriptions for each request then he/she send request and validate response against what is required and even test performance by checking latency of request.

After that developer/tester can automate a list of requests by giving each list excepted status of response and then all tests will execute to create coverage.

If coverage %100 that means all tests passed then Developer/Tester can generate documentations a first-generation called first commit to documentations after this commit the consumer of this API will notify and this doc will be available for him.

Also, admin will be aware of commits and versions, developer or tester can create a release after a set of commits when development reaches versions stage, each version includes release note about what new in this version.

The consumer of API will get the last version or last update of documentations to help hem keep update with the developer of the server and even platform create a to-do list of all endpoints in documentations. If the consumer connects with the endpoint he/she must check it in a to-do list to help admin to know about front end progress in case of API connection.

# Research Objectives

* Tool and service that ease the rest api testing and documentation
* Testing tool for API
* Making document for the test generated form it
* Making versions of that document
* Making collaboration more easy on the platform by providing a mean of chatting between them .
* Providing a snippet for the UI developer on how to link the API using different languages.

# Project type:

Application/service/tool.

# Project baseline requirements:

# Tools: software and hardware

## Software:  The software which is used in this project:

* React js .
* Node js express
* JetBrains PhpStorm.
* JetBrains WebStorm.
* Microsoft office word.

## Hardware requirement

* PCs
* Good internet connection

# Expected Outcomes

        Designing and implementing this system may help us in many ways: -

1. **Easing the testing process by automating it.**
2. **Simplifies the usage of the API for the developers by generating the proper documentation and offering templates to use**
3. **Better communication between the testers and the API developers (the team in general)**
4. **Improves some nonfunctional requirements of the API by testing them (Performance, Quality, Security...etc.).**
5. **Generally, saves time and effort**

**Literature review:**

**1-Curl [**<https://curl.haxx.se/docs/>**]:**

Command line tool and library for transferring data with URLs, allow you to test your API endpoints using command line.

Our platform has ability to test restful API endpoints using GUI rather than using command line, so using GUI make it easy to understands, no need to remember any command to achieve specific task.

Curl make request to server and get response only; our platform make using of request information and response to create documentation.

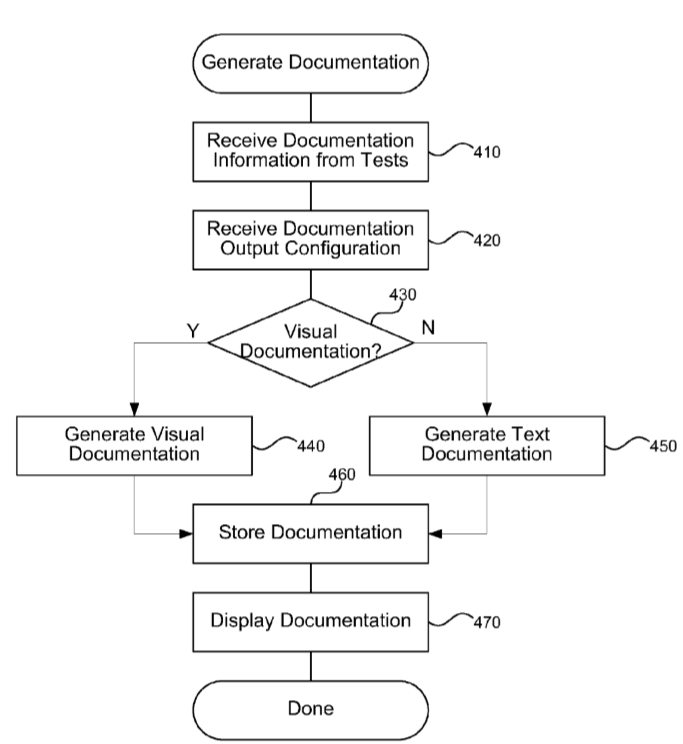
**2- Paper on GENERATING DOCUMENTATION FROM TESTS:**

**Intro:**

This tool receives test result for software code and generate documentation based on this test our platform provides the same idea but our documentation test for restful API end points and those end point represent result for functions in the server code.

Our platform allows documentations to have versions and release and keep this doc always update to development status by using ci/cd to docs.

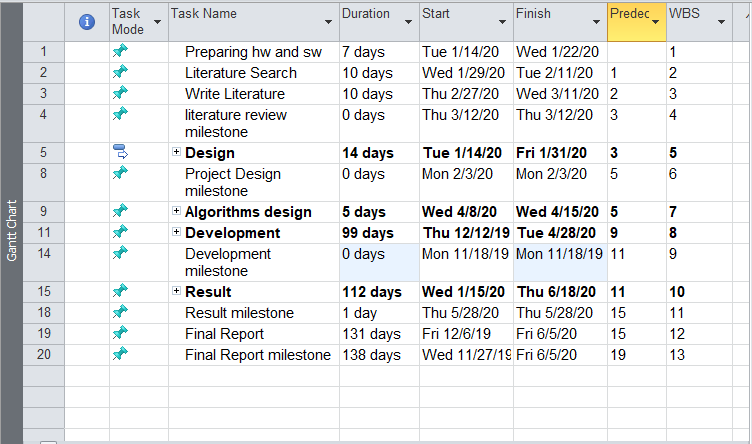
Even that our will doc includes helpers for stakeholders that will use documents in near future .to help them in API connection with client to improve productivity.



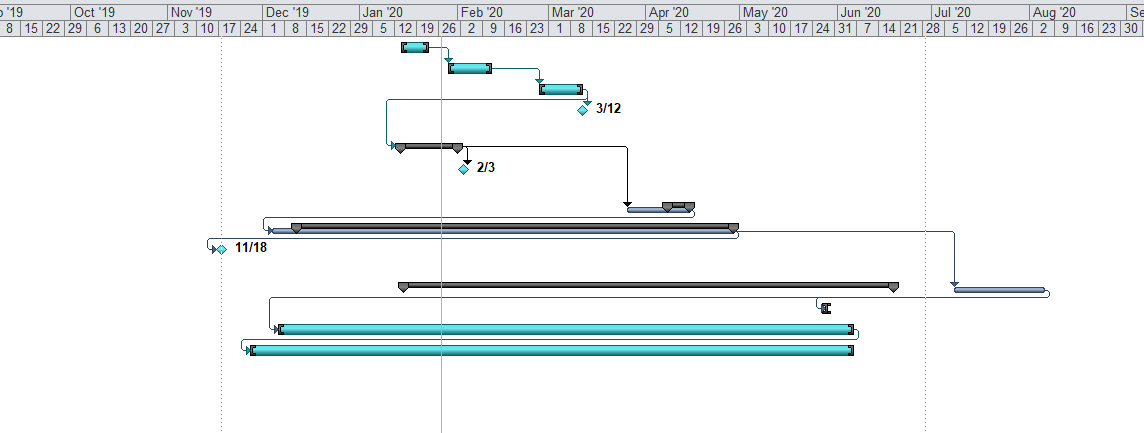
# Research Milestone

* Literature Review
* Project Design
* Development
* Result
* Final report

# Work breakdown structure:



Most of the system stages are interleaved; we can illustrate the milestone with Gantt chart. Here is the Gantt chart for this project:

****

# Risk analysis

|  |  |  |
| --- | --- | --- |
| **Risk** | **influence** | **solution** |
| **Absent dev** | **A** | **Continue on parts that doesn’t require him** |
| **Pc failure / OS crash** | **R** | **Resetting the pc** |
| **Loss of project file** | **g** | **Version control or multiple backups** |