

AN IDEA DOCUMENTAION FOR ACM Code2Create 2018

**PREDICT DISEASE AND THEIR SEVERITY BASED
ON 'INPUT SYMPTOMS' USING MACHINE LEARNING**

Submitted by,

TEAM - _pygeeks1991

Team Members,

1.Nabarun Chatterjee.

2.Palash Sarkar.

_____ *INSTITUTE* _____

**UNIVERSITY INSTITUTE OF TECHNOLOGY
THE UNIVERSITY OF BURDWAN
GOLAPBAG (NORTH), BURDWAN-713104.**

TABLE OF CONTENTS

TOPIC NO.	TITLE	PAGE NO.
1	IDEA STATEMENT	3
2	ABSTRACT	3
3	PROPOSED SOLUTION	3
4	TECHNOLOGY STACK AND SYSTEM MODULES	4
5	SCHEMATIC DIAGRAM	6

1.IDEA STATEMENT –

Predict disease and their severity based on ‘input symptoms’ using machine learning.

2.ABSTRACT –

Diseases have been a part of our lifestyle since ages. Whenever we are sick or suffer from some kind of ailment, we always turn to the doctors or some kind of medication specialist for the matter, thereby we end up spending a huge amount of money even if the problem was a small one. As times are passing, we are getting less and less scope to invest on routine checkups and healthcare. We often ignore a lot of small symptoms, thinking that they might not be worth the attention and that in-turn leads to something serious after a certain period of time.

3.PROPOSED SOLUTION –

What we propose is an intelligent system that can predict the disease/problem based on the symptoms, a particular person is suffering from. Not only this, the system will also be able give the severity of the disease and alert the person to consult a real-time doctor or get professional help. This will basically help people self-diagnose

and take pre-steps, towards their problem, which would in-turn defer from wrong treatment and other kinds of malpractice. The system will be trained on a detailed and well-structured data-set specifically designed and maintained for this kind of work. We'll train different machine learning models and take on different approaches to increase the efficiency and accuracy of the system. The system would show the list of 5 probable diseases related to the symptoms given, starting from the most probable one to the least. If the symptoms are new to the system, our application would crawl the web and gather data from reputed websites like 'webMD' and try to give an accurate result, along with training our model. The interface will be a website or a mobile app that'll be cloud controlled so that we can access all information whenever and wherever we require.

4. TECHNOLOGY STACK AND SYSTEM MODULES—

1. Model View Pages/Front-End Web Interface: *HTML, Bootstrap.CSS and DTL (Django template language) for dynamic content generation and responsive web design, for our website.*

2. Middleware for connectivity between server and web pages and also for DOM programming: *JavaScript, JQuery, AngularJS for async server connectivity and page content manipulation.*

3.Mainstream programming language and Web framework: *Python3 and Django web-framework for restful web services and request handlers and developing various computational modules.*

4.Web-crawling module: *Python3 along with Scrapy library for searching relevant information on the web.*

5.Machine-Learning module: *Python3 and ScikitLearn for the ai model (“**decision trees/random forests**”) for symptom analysis and disease prediction.*

6.NoSql Database for fast JSON formatting: *MongoDB for storing data-sets, results and other user related information.*

7.Mobile Application: *Android Studio with kotlin/java support for our mobile interface development.*

8.Deployment Server: *Apache Http Server or NGIX Http server for deploying the web application.*

9.Openshift Cloud Platform: *For pushing the local application docker image into the cloud with auto deployment/maintenance with Kubernetes.*

5.SCHEMATIC DIAGRAMS –

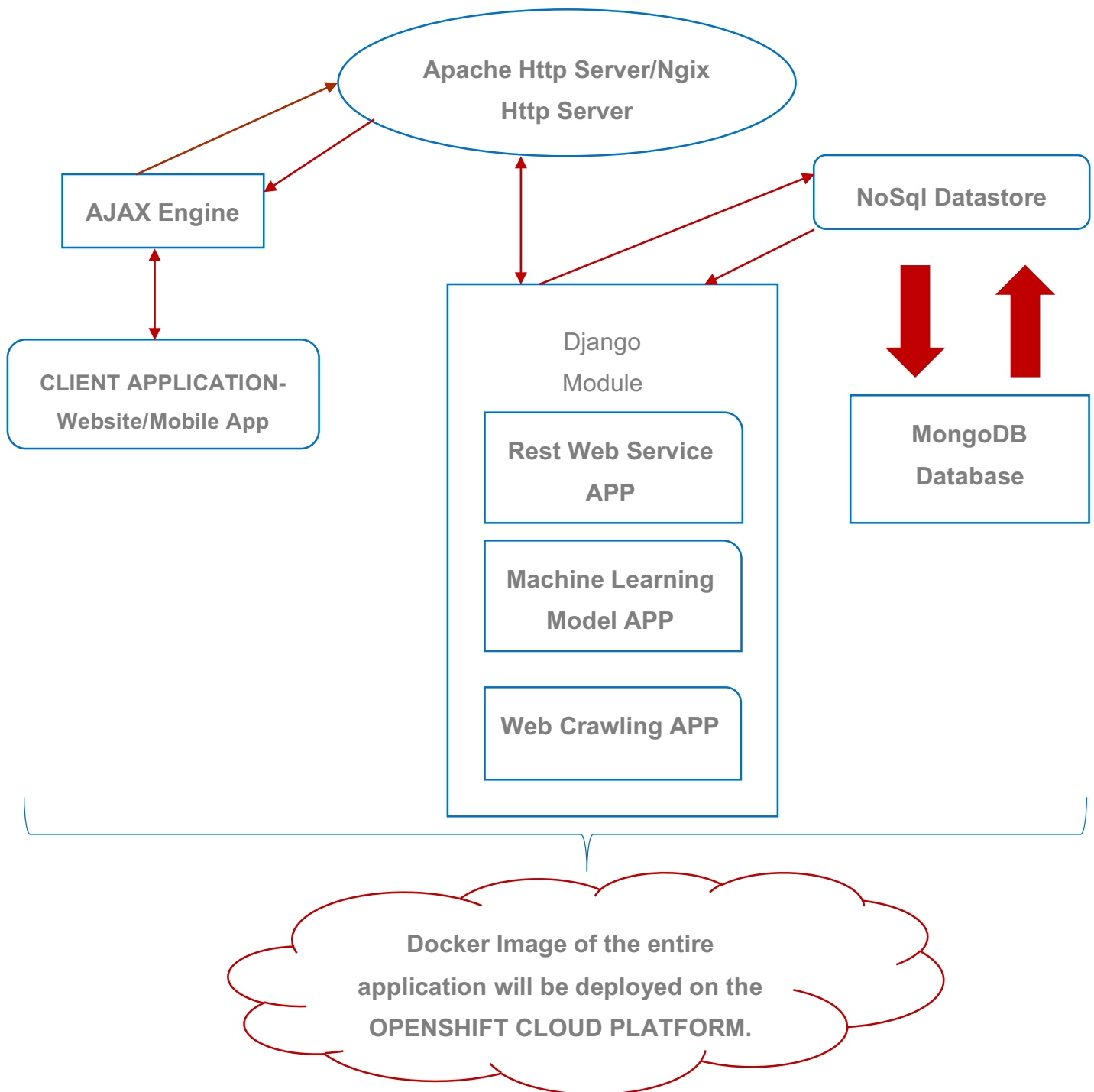


Figure 1: Overall 'Application Architecture' and 'Data-Flow'.

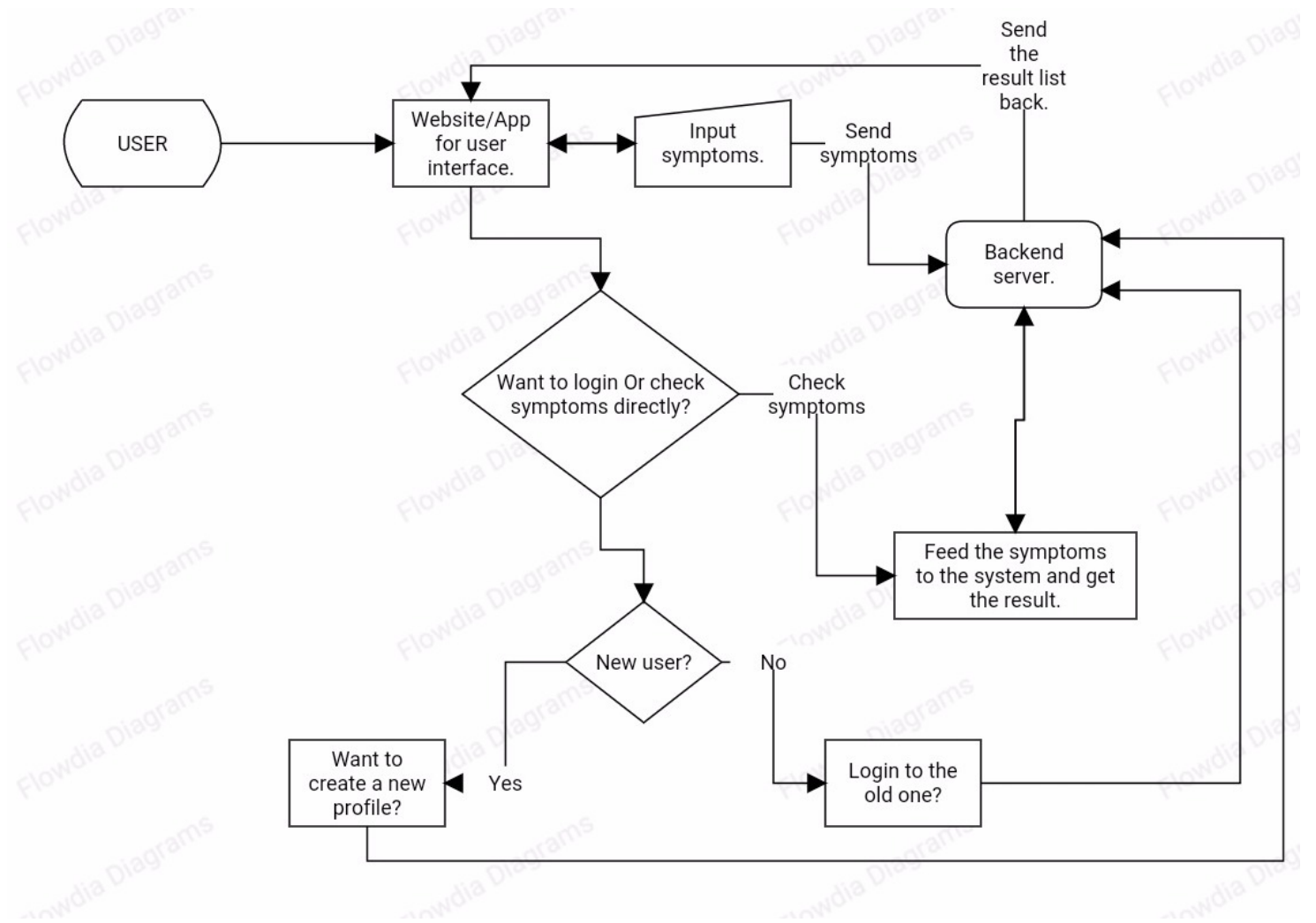


Figure 2: Internal logic structure and interaction with data.