International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 06 | June 2021 www.irjet.net p-ISSN: 2395-0072

Artificial Neural Network based Approach for Chronic Disease Prediction

Vineet Somwanshi[1], Dhiraj Phulaware[2], Sachin Bhatiya[3], Samkit Dodal[4]

[1-4]*School of Computer Science and Engineering, MIT World Peace University, Pune, India.*

Abstract - *The testing of blood samples by the patient is done with several traditional methods. There are variances in human-to-human manual test reports. This dissertation concentrates on a system aiming to reduce these issues. The most crucial in the world of medicine are efficient and accurate blood counts. Although the medical experts have huge ways available to get results suitably. This study aims to map the segmentation of the picture and the neural system in order to identify chronic disorders, such as Anaemia, high blood pressure and RBC, the WBC blood diffraction count of the patient. Until date, a single Haemocytometer slide test or automated Hemocytometer analysis was used for detecting blood cells. The advantages of both approaches, however, were merged with this approach to decrease time, money and effort and, on the top of the range, the output obtained tended to be more efficient and more accurate.*

Key Words: test reports, issues, medicine, chronic disorder, patient, accurate etc.

1. INTRODUCTION

The goal of this study is to detect anaemia in the human body by examining patient picture samples and hypertension via symptoms. The objective is to determine the kind of anaemia in the human body. The major emphasis of this initiative is anaemia and high blood pressure condition. This technology helps practitioners easily monitor the blood samples.

Blood samples are tested for any disease in the human body. The doctor provides additional therapy and drugs according to white blood cells and red blood cells ratio and symptoms. Anemia is attributed to a fall in blood hemoglobin levels under a set threshold dependent on the specific sex and age owing to reduced ability for transport oxygen. The structure of the red blood cells is modified due to low levels of hemoglobin. System compares the amount of WBC known as "lymphocyte" with the volumes of RBC in the blood sample picture.

An alternative approach of detecting the root cause of disease is historically used in the majority of pathological laboratories, combining the artificial Neural Network in the imaging process. The technology includes digital microscopic pictures of blood and input symptoms and estimates the average number of WBC and RBC cells. The technology recognizes illness type such as anaemia or hypertension on the basis of these two inputs and generates full documentation.

The system provided analyses the feasibility of an automated and reliable diagnosis of blood sample pictures and offers a technique for detecting the type of anaemia using light microscopic blood samples. Blood analysis is a significant technique for diagnosing these disorders. The Kohonen algorithm is used to assess the accurateness of the categorization of patterns of medical images linked to morphs of erythrocytes in the blood.

Blood samples are tested for any disease in the human body. The doctor provides additional therapy and drugs according to white blood cells and red blood cells ratio and symptoms. The practitioners now examine these samples with a microscope. If you do not have a very well-trained doctor or cannot recognize a blood cell aberration, you may not have a thorough blood testing report and the prescription of your doctor may create more health issues than anything good.

2. DESIGN

2.1 Class Diagram

The graph is a static graph. The static view of a request is represented. The class diagram is used not only to see, describe and record many components of a system, but also to build executable software application code. The class chart specifies the properties and functions of a class and also the system limitations.

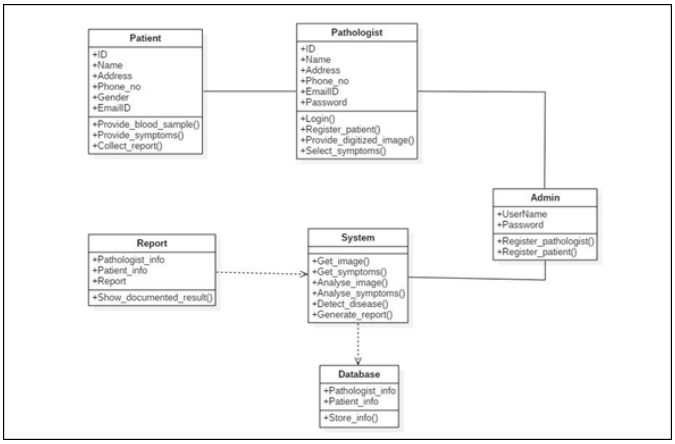

Fig -1: Class Diagram

2.2 State Diagram

One of the five diagrams that depict a system's dynamic nature with UML is state diagram. During their lifespan they

establish multiple object states and they are updated by events.

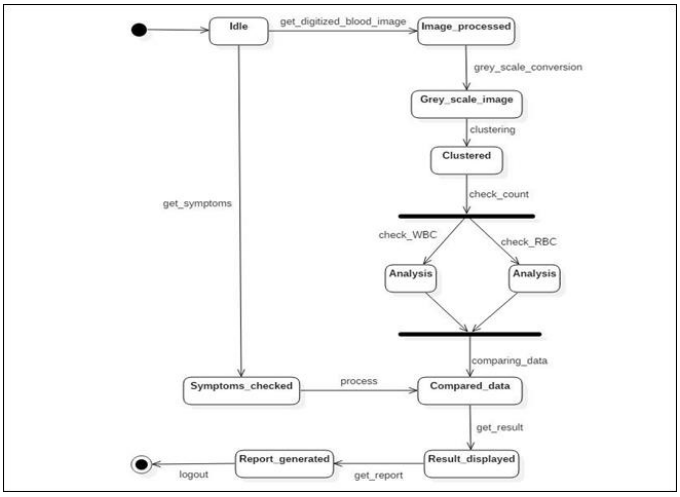

Fig -2: State Diagram

2.3 Sequence Diagram

Two diagrams known as the Sequence Diagram and the Collaboration Diagram depict this interactive behavior in UML. Both diagrams have identical fundamental functions. Sequence diagram highlights the temporal order and cooperation diagram stresses how the objects transmit and receive messages are structurally organized.

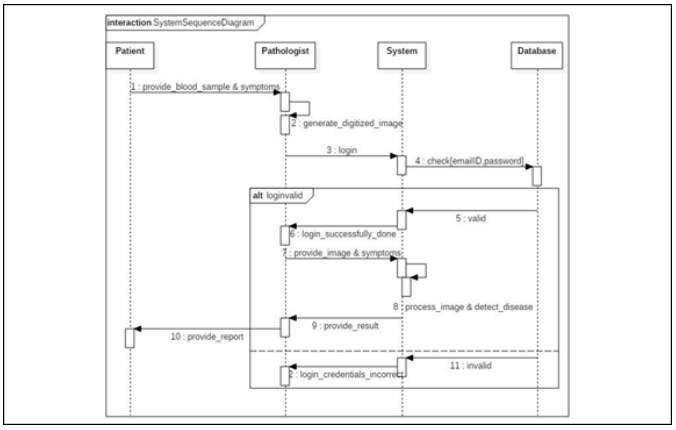

Fig -3: Class Diagram

2.4 Activity Diagrams

Basically, the activity diagram represents the flow from activity to activity. The operation can be defined as system operation. Thus, from one action to another the control flow is drawn. This might be sequential, connected or simultaneous. Activity diagrams address all types of flow control utilising various elements such as fork, connecting, etc.

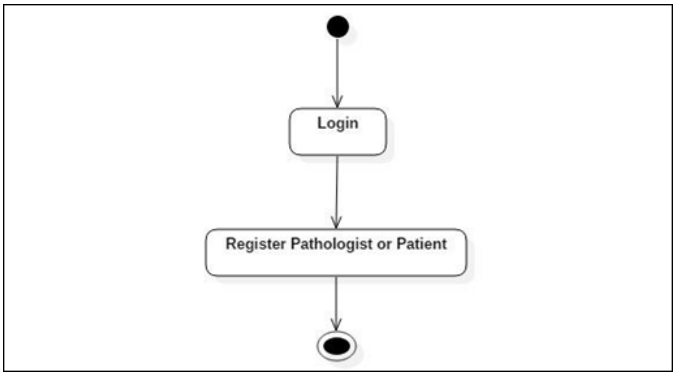

Fig -4: Admin Activity Diagram

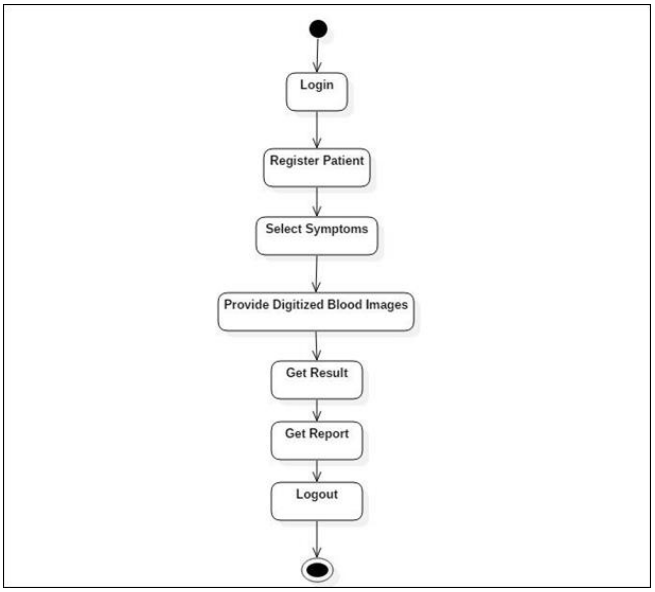

Fig -5: Pathologist Activity Diagram

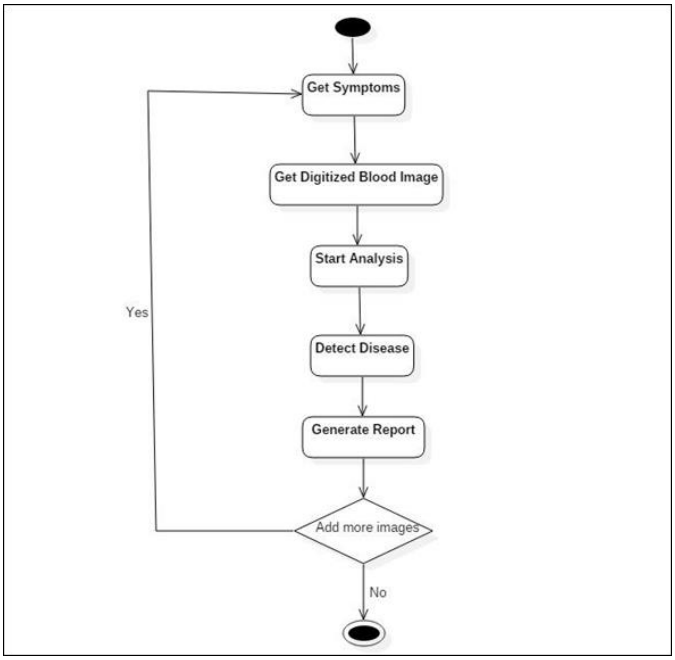

Fig -6: System Activity Diagram

2.5 Deployment Diagram

Deployment diagrams are used to depict the structure of a system with its physical components. This allows deployment diagrams to express a system's static deployment perspective.

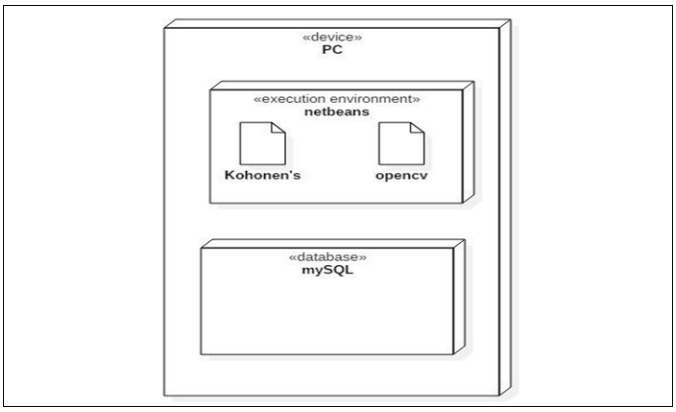

Fig -7: Deployment Diagram

3. IMPLEMENTATION

3.1 Grey Scale Conversions

You must generate 4 separate components of the file to generate a bitmap file. Bitmap header file, DIB header, color palette, raw bitmap information.

Bitmap File Header map contains:

i. Magic Number (2 bytes) at 0x0000 offset
ii. File Size (4 bytes) at 0x0002 offset
iii. Two 2 bytes long data, the values of which depend on the application that creates the file, at offsets 0x0006 and 0x0008
iv. Offset at which raw bitmap data can be found (4 bytes) at 0x000A

DIB header (Windows V3) contains:

i. Size of the DIB header (4 bytes) at 0x000E. Its value is 40, i.e. the size of the DIB header is 40 bytes.
ii. Width of the image in pixel (4 bytes) at 0x0012
iii. Height of the image in pixel (4 bytes) at 0x0016
iv. The number of color planes being used (2 bytes) at 0x001A. Its value must be set to 1.
v. Number of bits per pixel (2 bytes) at 0x001C. For the case of this particular project, its value is set to 8.
vi. Compression method being used (4 bytes) at 0x001E. In this case, the value is set to 0 which corresponds to no compression method being used.
vii. Size of the raw bitmap data (4 bytes) at 0x0022. More on this later.
viii. Horizontal and vertical resolutions of the image (pixels per meter). Each of these are 4 bytes and can be found at 0x0026 and 0x002A respectively. Their values are not really important.
ix. Number of colors in color palette (4 bytes) at 0x002E. In this particular project, 256 colors (shades of gray) are used. Therefore its value is set to 256.
x. Number of important colors used (4 bytes) at 0x0032. In this project, its value is set to 0 which corresponds to every color is important.

Color palette:

In this situation a 256-size color palettes [RGB(i,i,i) with 0 to 255] will be formed. One byte padding must be supplied at the end of every shade.

Raw bitmap data steps:

i. In raw bitmap data, pixels are mapped from left to right and from bottom to up.
ii. The left most pixel at the bottom of the bitmap image is stored first in raw bitmap data.
iii. At the end of each row of pixel necessary padding is added for 4 byte alignment, i.e the width of the bitmap file is divisible by 4 then no padding is added, if the remainder is 1, 2 or 3 then 3 byte, 2 byte or 1 byte padding is added respectively to maintain 4 byte alignment.
iv. For calculating the length of raw bitmap data, one has to calculate the number of byte padding required after each row. Here is how to calculate the size of raw bitmap data:
v. Raw data size (in bytes) = bitmap width* bitmap height*bytes per pixel + bitmap height* byte of padding at the end of each row
vi. The RGB to Gray scale is converted by using the following formula:
vii. Shade of gray = (int) (0.3 * R + 0.59 * G + 0.11 * B)
viii. Store converted image.
ix. Stop.

3.2 Grey Scale Image Scanning

Check Input:

Algorithm:

Step 1: Input Grey version of Blood Sample
Step 2: Convert image into X-Y Matrix with RGB values.
Step 3: Implement Hybrid SOM
Step 4: Generate and Print Report

Hybrid SOM:

- Analyze Image.
  i. Select Grey Scale Image
  ii. Convert into RGB matrix
  iii. Detect WBC and RBC images
- Generate two separate images of WBC & RBC based on their RGB values.
- Make two different Set s for RBC and WBC.
- For Each Cell detect store all (x,y) coordinates in separate Data Table.
- Calculate Size and number of WBC and RBC using distance learning algorithm (part of Hybrid SOM).

- Check one pixel at a time and traverse in all directions till white like pixel is detected.
- Each connected WBC and RBC pixels are stored in data tables and then stored in their respective Data Sets.
- Compare WBC and RBC set values to detect type of Anaemia.

3.3 Kohonen's Self-Organization Map (SOM)

The SOMs of Kohonen are a kind of unmonitored study. The objective is to uncover a certain underlying data structure. The SOM of Kohonen is known as a topology conservation map since the network nodes are subject to a topological structure. A topological map is only a neighborhood-preserving mapping. The geometry of the output nodes will be disregarded. In a layer every node is the same since every node in the higher or lower layer is linked to all of the nodes. This is why these nodes are physically arranged. Closer nodes interact differently than nodes that are "far apart." They interact differently. What's near to and far? What's far? The output nodes can be organized in a line or in a planar setup.

The objective is to train the net to match neighboring outputs. For example if x1 and x2 are two input vectors and t1 and t2 are the winning output node position, then t1 and t2 should be close if x1 and x2 are comparable. The feature map is called a network that carries out this sort of mapping. Neurons in the brain tend to clump together. The connections inside the group are far larger than the neurons outside the group. This is easily imitated by the Kohonen network.

Algorithm for Kohonon's Self Organizing Map:

- Assume output nodes are connected in an array (usually 1 or 2 dimensional)
- Assume that the network is fully connected - all nodes in input layer are connected to all nodes in output layer.
- Use the competitive learning algorithm as follows:
- Randomly choose an input vector x
- Determine the "winning" output node i, where wi is the weight vector connecting the inputs to output node 1.
- Note: the above equation is equivalent to wi x >= wk x only if the weights are normalized.

$$\left|\omega_i - x\right| \le \left|\omega_k - x\right| \qquad \forall k$$

- Given the winning node i, the weight update is;

$$w_k(new) = w_k(old) + \mu\, \aleph(i,k)\,(x - w_k)$$

When i=k falls off with the |rk — ri | unit I and k distance from the output array, this neighborhood function is invoked. Units in the victory and in the winner themselves are therefore noticeably upgraded in their weights. The weights of remote output nodes do not considerably change. The topological information is provided here. Close units receive comparable updates and hence react to close input models.

The above rule drags the weight vector wi and the weights of nearby units towards the input x.

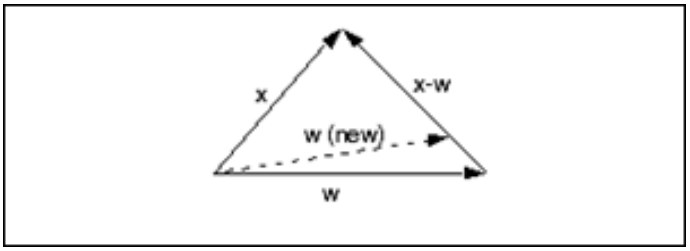

Fig -8: Weight vector transfer

Example of the neighborhood function is;

$$\aleph(i,k) = e^{\left(-\left|r_k - r_i\right|^2\right)/(2\sigma^2)}$$

SOM Hybrid:

The basic technique of SOM (Self organizing maps) is to collect a random point of the x-y axis and evaluate the RGB values of a respective point similarly by finding the RGB values for other random items and storing the object in a table in order to discover the required object.

But the standard SOM is not applicable here since this can only apply to photos in a defined format of a certain pattern. We know, however, that pictures of the blood samples change dramatically according on the staining component. The RGB values therefore vary considerably as well.

A broad variety of RGB values is available in the image. Further assessments get more complex and vast changes are achieved in RGB values, which mean that we can now turn the real image into a gray scale, avoiding dramatic fluctuations in RGB values and readily distinguishing RBC and WBC. As the RGB values stay consistent in grey scales, it is now straightforward to figure out the range of RGB values for the RBC, WBC and the backdrop and these three constituents may now be separated between themselves. Therefore, two pictures are taken, one from RBC alone, and one from WBC alone. SOM hybrid basically uses the distance-learning algorithm to separate the RBC's and the WBC's.

Distance Learning Algorithm:

The distance learning algorithm is a component of SOM, which essentially helps in the separation between the RBC and the WBC from the remainder.

- Plot the image onto positive x and positive y-axis, hence any point (that is a pixel) on the image contributes some (+x, +y) coordinates.
- Choose random points on the image
- Now the RGB value of the respective point will be either same as RBC, WBC or the background.
- Now the point will move in all the four directions that is +x, -x, +y, -y directions where the movement

will be constrained to single pixel (that is Distance = 1 pixel) till the RGB values obtaining changes.
- Recursively performing the above action.
- Obtaining all such values and storing in a table, each table constitutes to a single cell.
- Maintaining a data set that can be defined as collection of tables (cells).
- All RBC's will form one data set according to its predefined RGB range.
- Similarly all WBC's will form one data set according to its predefined RGB range.
- Thereby separating RBC, WBC and the background.

The results are therefore made easier in the form of two photographs, one of RBC and the other of solely WBC's.

4. TEST PLAN AND RESULTS

In order to uncover difficulties, the test aims to verify, validate and discover errors. And these difficulties are to be found in order to resolve them.

Verification is the verification or testing, by comparing findings against pre-specified demands, of the things including software for conformity and consistency.

Error detection: Testing should purposely try to make things go wrong to see if things should occur when it should not or when things should not occur.

The validation examines system accuracy – i.e. the procedure to ensure if the user truly requested what was stated.

Few points considered during testing were:

- Testing should be carried out with a purpose of finding errors rather than to determine what is correct.
- Good test cases should be used.
- Line by line code walkthrough and inspection should be carried out.
- Checking number of actual and formal arguments of a function to be in agreement.
- Unnecessary code should be removed.
- Proper validation of the fields should be carried out.
- Testing should be carried out in structured manner starting from the testing of each module to that of the testing of the GUI.

Testing is performed at the following levels:

i. Unit testing

ii. Integration testing

iii. System testing

Unit Testing:

The smallest software part is tested alone in this manner. Unit testing is used to validate the proper functioning of the particular source code unit.

Unit testing is a software design and development process in computer programming where the programmer checks if individual source code units function correctly. A unit is an application's smallest testable element. A unit may be an individual programme, function and procedure in procedural programming, but the smallest unit in object oriented programming is a method that may be part of a base/super class, abstract or derived/child class. Unit testing is typically done by software developers to ensure that the code other developers have written meets software requirements and behaves as the developer intended.

Integration Testing:

This methodology includes the testing of the integrated system or subsystem groups of composite modules. It concentrates on software architecture design and construction with control of the accuracy of the interfaces between modules. Integration testing is easier when the unit testing is performed extensively and there are clearly defined interfaces between components.

System Testing:

System tests must be performed on a whole integrated system to assess compliance of the system with its set requirements. In "inter-assemblies" and the whole system, both problems appeared to be detected.

5. SYSTEM REQUIREMENTS AND SPECCIFICATIONS

5.1 Minimum System Requirements

- Minimum free 512MB Hard Drive space.
- Minimum 512MB RAM.
- Intel Dual Core Processor or faster.
- Operating System Windows XP onwards.
- Java Supported.
- To run the project we require at least of the above mention requirements.

6. CONCLUSIONS

From the above study, conclusions drawn are;

i. Blood samples are tested to detect any disease in the human body. The doctor provides additional therapy and drugs according to white blood cells and red blood cells ratio and symptoms. Anemia is attributed to a fall in blood hemoglobin levels under a set threshold dependent on the specific sex and age owing to reduced ability for transport oxygen.

ii. The "Artificial Neural Network Based Approach for Chronic Disease Prediction" examines your picture in the blood sample to identify anaemia in your human body. The approach identifies the three forms of anaemia and hypertension as Macrocytic, Normocytic and Microcytic.

iii. The difference between the volume of RBC and WBC known as the Lymphocyte is compared to diagnose anaemia type. We used an application called as COLOR PIC to know the range of RGB values of various ingredients in blood sample images. Hybrid SOM and the Distance Learning Algorithm were used to separate the WBC and RBC, and also to compare the amount of RBC and WBC.

ACKNOWLEDGEMENT

We thank Professor Pratvina Talele, Department of Computer Science and Engineering, MIT World Peace University, for assistance and comments that greatly improved the manuscript.

REFERENCES

[1] "A Condition Monitoring Algorithm Based on Image Geometric Analysis for Substation Switch" Authors: - Jicang Lu, Hui Lin, Weizheng Zhang and Xiaowei Shi M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[2] "A case-based classifier for hypertension detection, Knowledge-Based Systems" Authors: - K. Hsu, C. Chiu, N. Chiu, P. Lee, W. Chiu, T. Liu, and C. Hwang.

[3] "An Adaptive Framework for Personalized Recommendation Algorithms" Authors: - Jianchang Tang and Xinhuai Tang.

[4] "A Logistic Regression and Artificial Neural Network-based Approach for Chronic Disease Prediction: a Case Study of Hypertension" Authors: - Aiguo Wang, Ning An, Yu Xia, Lian Li and Guilin Chen.

[5] "Neural Networks" (Algorithms, Applications, Programming, Technology) Saues AcFree Man, David M Skapura, Pearson Publisher 2009 Edition.

[6] Chen P, Yan X, Zheng C. Recognition of Blood and Bone Marrow Cells using Kernel- based Image Retrieval. IJCSNS 'International Journal of Computer Science and Network Security" 2006; 6:29-34.

[7] Szolovits P, Patil RS, Schwartz WB. Artificial intelligence in medical applications. Annals of International Medicine 1988; 108:80-8.

[8] "Advanced Medical Statistics" Editors Ying Lu, Ji- Qian Fang Published By World Scientific Publishing Co. Pte. Ltd, 5 Toh Tuck Link Singapore 596224.

[9] "Software Testing" by Ron Patton Second Edition Published by Dorling Kindersley (India) Pvt. Ltd, licensed of Pearson Education in South Asia.

BIOGRAPHIES

Vineet Somwanshi
Department of Computer Science and Engineering,
MIT World Peace University.

Dhiraj Phulaware
Department of Computer Science and Engineering,
MIT World Peace University.

Sachin Bhatiya
Department of Computer Science and Engineering,
MIT World Peace University.

Samkit Dodal
Department of Computer Science and Engineering,
MIT World Peace University.