**CHAPTER 4**

**DEVELOPMENT PROCESS**

* 1. **REQUIREMENT ANALYSIS**

Requirements are a feature of a system or description of something that the system is capable of doing in order to fulfil the system’s purpose. It provides the appropriate mechanism for understanding what the customer wants, analyzing the needs assessing feasibility, negotiating a reasonable solution, specifying the solution unambiguously, validating the specification and managing the requirements as they are translated into an operational system.

* + 1. **PYTHON:**

Python is a dynamic, high level, free open source and interpreted programming language. It supports object-oriented programming as well as procedural oriented programming. In Python, we don’t need to declare the type of variable because it is a dynamically typed language.

For example, x=10. Here, x can be anything such as String, int, etc.

Python is an interpreted, object-oriented programming language similar to PERL, that has gained popularity because of its clear [syntax](https://whatis.techtarget.com/definition/syntax) and readability. Python is said to be relatively easy to learn and portable, meaning its statements can be interpreted in a number of [operating system](https://whatis.techtarget.com/definition/operating-system-OS)s, including UNIX-based systems, Mac OS, MS-DOS, OS/2, and various versions of Microsoft Windows 98. Python was created by Guido van Rossum, a former resident of the Netherlands, whose favourite comedy group at the time was Monty Python's Flying Circus. The source code is freely available and open for modification and reuse. Python has a significant number of users.

**Features in Python**

There are many features in Python, some of which are discussed below

* Easy to code
* Free and Open Source
* Object-Oriented Language
* GUI Programming Support
* High-Level Language
* Extensible feature
* Python is Portable language
* Python is Integrated language
* Interpreted Language
  1. **ANACONDA**

Anaconda distribution comes with over 250 packages automatically installed, and over 7,500 additional open-source packages can be installed from [PyPI](https://en.wikipedia.org/wiki/Python_Package_Index) as well as the [conda](https://en.wikipedia.org/wiki/Conda_(package_manager)) package and virtual environment manager. It also includes a GUI, Anaconda Navigator,[[12]](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)" \l "cite_note-12) as a graphical alternative to the command line interface (CLI).

The big difference between conda and the [pip package manager](https://en.wikipedia.org/wiki/Pip_(package_manager)) is in how package dependencies are managed, which is a significant challenge for Python data science and the reason conda exists.

When pip installs a package, it automatically installs any dependent Python packages without checking if these conflict with previously installed packages. It will install a package and any of its dependencies regardless of the state of the existing installation. Because of this, a user with a working installation of, for example, Google Tensorflow, can find that it stops working having used pip to install a different package that requires a different version of the dependent NumPy library than the one used by Tensorflow. In some cases, the package may appear to work but produce different results in detail.

In contrast, conda analyses the current environment including everything currently installed, and, together with any version limitations specified (e.g. the user may wish to have Tensorflow version 2,0 or higher), works out how to install a compatible set of dependencies, and shows a warning if this cannot be done.

Opensource packages can be individually installed from the Anaconda repository, Anaconda Cloud (anaconda.org), or the user's own private repository or mirror, using the conda install command. Anaconda, Inc. compiles and builds the packages available in the Anaconda repository itself, and provides binaries for Windows 32/64 bit, Linux 64 bit and MacOS 64-bit. Anything available on [PyPI](https://en.wikipedia.org/wiki/Python_Package_Index) may be installed into a conda environment using pip, and conda will keep track of what it has installed itself and what pip has installed.

Custom packages can be made using the conda build command, and can be shared with others by uploading them to Anaconda Cloud, [PyPI](https://en.wikipedia.org/wiki/Python_Package_Index) or other repositories.

The default installation of Anaconda2 includes Python 2.7 and Anaconda3 includes Python 3.7. However, it is possible to create new environments that include any version of Python packaged with conda.

### Anaconda Navigator

Anaconda Navigator is a desktop [graphical user interface (GUI)](https://en.wikipedia.org/wiki/Graphical_user_interface) included in Anaconda distribution that allows users to launch applications and manage conda packages, environments and channels without using [command-line commands](https://en.wikipedia.org/wiki/Command-line_interface). Navigator can search for packages on Anaconda Cloud or in a local Anaconda Repository, install them in an environment, run the packages and update them. It is available for [Windows](https://en.wikipedia.org/wiki/Windows), [macOS](https://en.wikipedia.org/wiki/MacOS) and [Linux](https://en.wikipedia.org/wiki/Linux).

The following applications are available by default in Navigator:

* [JupyterLab](https://en.wikipedia.org/wiki/Project_Jupyter#JupyterLab)
* [Jupyter Notebook](https://en.wikipedia.org/wiki/Project_Jupyter#Jupyter_Notebook)
* QtConsole
* [Spyder](https://en.wikipedia.org/wiki/Spyder_(software))
* [Glue](https://en.wikipedia.org/wiki/Glue_(software))
* [Orange](https://en.wikipedia.org/wiki/Orange_(software))
* [RStudio](https://en.wikipedia.org/wiki/RStudio)
* [Visual Studio Code](https://en.wikipedia.org/wiki/Visual_Studio_Code)
  1. **RESOURCE REQUIREMENTS:**
     1. **SOFTWARE REQUIREMENTS**:

|  |  |
| --- | --- |
| Operating System | Windows 7or later |
| Simulation Tool | Anaconda (Jupyter notebook) |
| Documentation | Ms – Office |
|  |  |

* + 1. **HARDWARE REQUIREMENTS:**

|  |  |
| --- | --- |
| CPU type | I5 |
| Ram size | 4GB |
| Hard disk capacity | 80 GB |
| Keyboard type | Internet keyboard |
| Monitor type | 15 Inch colour monitor |
| CD -drive type | 52xmax |

* 1. **ARCHITECTURE**

**PROPOSED SYSTEM:**

**Data collection (images and xml file)**

**Data pre processing**

**Model implementation (OpenCV)**

**Implementing into tkinter frame work**

**Final prediction**

**Select an image**

**Select a xml file**

**Segmented WBC and RBC**

**Histogram plot values for RBC and WBC**

* In this proposed Work, we have implemented graphical user interface (GUI).
* We use blob algorithm for image processing .It will filter the colors from the image then it will begin segmentation of the image
* Detecting a blob (A Blob is a group of connected pixels in an image that share some common property)
* OpenCV provides a convenient way to detect blobs and filter them based on different characteristics using its built-in function called Simple Blob Detector.
* Its algorithm is controlled by parameters in terms of Color, Size and Shape where Shape parameter has three specific parameters: Circularity, Convexity and Inertia

**ADVANTAGES**

* We have implemented graphical user interface (GUI) which make the user to easily interact with the application
* The implementation open CV for image processing and analysis for the platelet, red blood and white blood cells was made possible and resulted to high level of accuracy
* The algorithm was able to meet the margin of error of less than 10%. Furthermore, improved accessibility and portability of the software used were obtained through its on-line database support and free ware feature.
* Here we are use xml files for storing and retrieving data for portability and convenient to reduce the memory
  1. **SYSTEM MODULES**

MODULE 1: Dataset collection

MODULE 2: Data preprocessing

MODULE 3: Model implementation

MODULE 4: Tkinter frame work

MODULE 5: Prediction

**Module 1: Data Collection**

* Data collection allows you to capture a record of past events so that we can use data analysis to find recurring patterns. From those patterns, you build predictive models using machine learning algorithms that look for trends and predict future changes.
* The main purpose of data collection is to gather information in a measured and systematic manner to ensure accuracy and facilitate data analysis. Since the data collected is meant to provide content for data analysis, the information gathered must be of the highest quality for it to be of value.
* Here for this project we are using image

**Module 2: Pre processing**

* Information pre-processing could be a portion of information mining, which includes changing crude information into a more coherent organize.
* Data preprocessing may be a information mining method which is utilized to convert the crude information in a valuable and effective format.
* Steps Included in Information Preprocessing: Data cleaning
* **Module 3: model implementation**
* Here for this project we are using opencv model to implement in this project.
* OPENCV:
* OpenCV is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today’s systems. By using it, one can process images and videos to identify objects, faces, or even handwriting of a human. When it integrated with various libraries, such as NumPy, python is capable of processing the OpenCV array structure for analysis. To Identify image pattern and its various features we use vector space and perform mathematical operations on these features.
* **Module 4: TKINTER FRAME WORK**
* Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.
* Creating a GUI application using Tkinter is an easy task. All you need to do is perform the following steps −
* Import the *Tkinter* module.
* Create the GUI application main window.
* Add one or more of the above-mentioned widgets to the GUI application.
* Enter the main event loop to take action against each event triggered by the user.

.

* **MODULE 5: Prediction**

Finding the white blood cells, red blood cell.

In the given dataset we are finding the white blood cells and red blood cells.