MAL.ARIA
IDENTIFICATION MADE SIMPLE

# Assessing malaria blood samples project objectives and implementation

## Minimal viable product objectives (W11)

An image classification model for detecting malaria in red blood cells using convolutional neural networks. When testing the model, the accuracy should be reasonably high. It should have a front-end user interface using web development, however at this point it's most likely to be quite simplistic. For example, the front end will have an image input, followed by an output of whether the sample is infected using the model.

| Specific MVP milestones | Estimated start/end dates |
|---|---|
| Create and train a convolutional neural network using python that identifies whether a red blood cell is infected. | 19/10/20 - 16/11/20 |
| Test the CNN and adjust accordingly to increase accuracy and remove unfavorable features such as overfitting. | 16/11/20 - 23/11/20 |
| Connect the algorithm to a local host web page. At this point it should take an image input and return whether the sample is infected. | 23/11/20 – MVP Presentation |

## End of academic year objectives

An image classification model to detect malaria in red blood cells using convolutional neural networks, which consists of many different hidden layers and other types of image processing. There will be a stylish, intuitive user interface, implemented using web development, that is simplistic to operate from the user's perspective. Additionally, note that I will specifically implement the algorithm using a website instead of an android app, so it can reach a wider audience. This is vital, as it is aimed at poorer areas of the world. The home UI will consist of an image input selector and a step-by-step detailed explanation on how the system works. The inputted image will be processed and passed through the model to decide whether the sample is infected. The output will be presented with the relevant information on what to do next if infected.

## Extra objectives

- Implement an intuitive front end to my algorithm that can be used on numerous types of devices. The best option seems to be a website using bootstrap. I believe this is imperative, as in poorer areas it needs to be readily available with the minimum number of resources.
- Explore computer vision deep learning techniques for image classification that will be beneficial to my project.
- The model should be accurate and produce a high accuracy from testing of true positives/negatives. This is crucial because false positives and critically false negatives should not occur for a project of this kind. It can be a matter of life or death. I will also provide bold disclaimers, as any image classification model will make mistakes.
- Although it is not a critical feature in relation to the model, I will aim for an aesthetically pleasing user interface. It should be simple and easy to use with a smart identity. Research suggests that users feel they can trust a modernized website.

## Implementation tools

- Research had directed me towards using python as there are lots of useful libraries when creating convolutional neural networks and machine learning in general. The libraries for creating the convolutional neural network which seem the most beneficial are TensorFlow and Keras. Similarly, there are many other libraries that will be useful in aiding the network design.
- When designing the front end to my project I am planning to create a website. This will implement libraries like bootstrap so my tool can be used across any device which has access to the internet.