Technology Selection Rationale for UWAM FRACAS

The group has decided on a framework of HTML/CSS/React.js frontend with a Python/Django/PostgreSQL and to use Amazon Web Services (AWS) for server purposes.

HTML, CSS, and JavaScript are necessary to build a webpage of any capacity. For HTML and CSS, we will utilise the most recent versions, HTML5 and CSS3. For JavaScript we have decided to use the React library due to it being prominent in the industry and more coder-friendly than vanilla JavaScript.

Python was the only backend language considered as it is the preferred language of the entire team as well as the language the team has the most experience with. For using Python on the backend, the common frameworks are Flask and Django. Flask was considered as some members have experience with it. However, the team decided on Django due to Django's inbuilt database capabilities which should save time by alleviating SQL coding. As for a data storage a relational database was preferred over non-relational options such as MongoDB or DynamoDB; this is due to the client's specification that if they decide in the future to add attachments to failure reports it will be implemented by hosting the file elsewhere and adding a link to the file in the report. MySQL and PostgreSQL were considered, but PostgreSQL was chosen as it is supports more complex query capabilities, ideal for futureproofing. PostgreSQL is also supported by Amazon RDS.

After a fruitless attempt to locate a viable in-house server at The University of Western Australia (UWA) the decision was made to use Amazon Web Services (AWS) for hosting purposes while in testing and development. The goal is to stay within AWS's free tier while in development and once the project is completed it can be deployed elsewhere as necessary. AWS Elastic Cloud Compute (EC2) will be used for the backend and RDS for the database configuration. Amazon Virtual Private Cloud (VPC) will be used to create security groups among other security measures.

Testing: A variety of testing suites and libraries will be employed to mitigate risks of abnormal behaviour, bugs, and defects. These include Pytest for Python, UnitTest for Django, Jest and React Testing Library for React and JavaScript, and finally Selenium for browser-based testing for HTML and CSS.