Solihull College and university centre wishes to procure a "Service Status Monitoring Solution" to allow stakeholders to quickly monitor the status of services critical to the delivery of teaching and learning and business as usual.

**Suggested Technical specs:**

* The system should be web based
* Should be capable of presenting a clear user interface with a traffic light system against listed services to depict service status
* Perform tests to ascertain the service status and uptime of configurable services using ping and http requests
* Be capable of querying the College's MySQL based ticketing system to display service affecting problems to inform users of known 'problems'
* Be able to report uptime of monitored services.

**Activity A**

Produce a detailed proposal for a digital solution that you would develop to meet the

needs of the client.

Your proposal should provide a rationale for the solution you are proposing and include:

• the business context

• the functional and non-functional requirements of the solution

• decomposition of the problems that will need to be solved to implement the

functional and non-functional requirements

• the key performance indicators (KPIs) and user acceptance criteria for the

proposed solution

• a description of the proposed solution

Justification of:

• how the recommended solution meets the needs of the client and users

• how potential risks will be mitigated

• What relevant regulatory guidelines and legal requirements, in relation to software

development and the education sector, will be addressed

test plan and strategy

what test you will use and why

**Activity B**

Produce a set of design documents for the digital solution you are proposing. The design

documentation must include:

• **visual/interface designs** – wireframes and styles guides

• **data requirements** – what data is required by the site to operate, data dictionary

• a selection of **algorithm designs – flow charts on the most important parts of the software**

**• a test strategy – what, where and why, integration, unit,** fix bugs and document

Your proposed solution must be of sufficient scope and complexity to demonstrate your

ability to implement code in at least two appropriate languages to implement front-end

and back-end processes.

Your design documents should provide sufficient detail to:

• effectively communicate the intended solution to both technical and

non-technical stakeholders

• allow the client to make informed decisions

• allow a third-party developer to use the design documents to create the

proposed solution.

Your test strategy does not need to detail specific tests for every part of the solution.

However, it should provide a selection of tests, relevant to your proposed solution,

that show:

• the order in which you intend to test all components of the solution

• the types of test to be carried out for each component.

Your algorithm designs do not need to show the whole solution but should show how

you would solve a number of different key problems. You should limit your algorithm

designs to a maximum of five complex problems.

**W3 accessibility**

**Activity C -** Developing the prototype

Develop a functional prototype of your proposed digital solution to meet the

client’s needs.

During development, make sure you:

• implement secure code in at least two appropriate languages to implement front-end

and back-end processes

• gather and prepare appropriate assets to be used in the development of your

digital solution

• make use of, and document, iterative testing

• document the iterative development process, including changes made to the

solution during development

• produce code that can be maintained by a third party

• ensure a high-quality user experience

• follow legal and regulatory guidelines and standards.

When gathering assets, you need to record the sources you use in an assets log.

Your log must:

• record all the sources you have used

• describe the content and its intended purpose

• log the date on which you retrieved the information.

Your testing documentation does not need to include details of every test carried out.

However, it should demonstrate:

• an iterative approach to testing

• understanding of how to test inputs, calculations, validation and processes using

appropriate test data