Criterion	7	6	5	4	3	2-1
API Endpoint functionality, error responses and application reliability. 35 + 15 = 50 marks	The server successfully implements all of the API endpoints according to their Swagger doc specifications, including authenticated search and time-based filtering. The application is robust and executes without noticeable error and handles service failures and errors gracefully. Error conditions are returned in accordance with the API spec and response codes are perfectly according to specification.	The server supports all of the API endpoints and JWT authentication, but there are some limitations in the implementation. [See discussion of 6-7 levels in the spec.] The application is robust and executes without noticeable error and handles service failures and errors gracefully. Error conditions are mostly returned in accordance with the API spec, with perhaps occasional incorrect response codes.	The server supports some API endpoints to 5 level, with limitations in filtering and authentication. [See discussion of 5 level in the spec.] The application is mostly robust and usually executes without error. Service failures and errors are handled adequately. Error conditions are basically in accordance with the API spec, but there are may be a number of incorrect response codes.	The server supports some API functionality to 4 level, limited filtering and no authentication. [See discussion of 4 level in the spec.] The application basically works, but the robustness is questionable and there are obvious errors and service failures are not handled well. Error conditions are often not in accordance with the API spec	A basic express app with some functioning routes is presented, but the API is markedly incomplete or poorly implemented. The application basically works, but there are numerous errors and failures. The API error responses are not to specification.	The server does not meet the specification or is fundamentally flawed. The application is markedly incomplete and results in numerous errors.
Application Architecture and Middleware – DB Connectivity, Logging and Security. 20 marks	The application architecture is professional and uncluttered. The routes are well-organised and handled by appropriate routers. There is good use of middleware for DB connectivity, logging and security. There is no use of raw SQL. Self-signed HTTPS deployment is successfully demonstrated. User credentials are handled professionally.	The application architecture is reasonably professional and uncluttered. The routes are well-organised and routers are usually organised appropriately. There is good use of middleware for DB connectivity, logging and security, but there may be some variations in the approach. There is no use of raw SQL. Self-signed HTTPS deployment is successfully demonstrated. User credentials are handled professionally.	The application architecture is good but some aspects are awkward, with some routes not handled especially well, and routers not split appropriately. There is some use of middleware, but this is not as clearly thought out as it needs to be. There may be minor use of raw SQL. .Self-signed HTTPS deployment may not have been achieved. There may be some inappropriate handling of credentials.	The application architecture is basically sound but the route organisation is very clumsy in parts, and there is too much functionality handled at the application level or in large routers. There is limited use of middleware. There may be more extensive use of raw SQL. Self-signed HTTPS deployment is not attempted. Hashing of passwords is poorly implemented.	The application architecture is not well chosen, the routes are not well organised and there is little or no use of middleware. Raw SQL may be common. Security is poor and HTTPS is not attempted. Hashing of passwords is not attempted.	The application architecture is flawed and there is no use of middleware. Security is deeply flawed or not attempted.
(70 Marks)						

Development Process & Code Quality 10 marks	The application shows clear evidence of a professional approach to development, with a coherent modular structure and code quality at a near professional level.	The application shows good evidence of a professional approach to development, with a coherent modular structure and code quality at a near professional level, apart from some minor variations.	The application shows some evidence of a professional approach to development, with a reasonably coherent modular structure and competent, but not always professional level code quality.	The application shows limited evidence of a professional approach to development, with ad hoc or unclear organisation of the application and variable levels of code quality.	Application development has been ad hoc and little more than hacking, with no obvious organisation. Code exhibits numerous defects when compared to the standard expected.	Application development is deeply flawed, with little structure and poor code quality.
Report and User Guide 20 marks	The report is thoroughly professional and addresses each of the listed requirements in detail and with only occasional errors of grammar or structure.	The report is thoroughly professional and addresses each of the listed requirements in detail and with some more frequent errors in grammar or structure.	The report is professional but lacks some detail in a small number of the listed requirements. There may be occasional errors of grammar or structure.	The report is adequate, but the coverage is deficient in a number of the listed requirements. Grammar and structure may be somewhat variable, but are overall ok.	The report is somewhat adequate, but the coverage is deficient in many of the listed requirements. Grammar and structure are of variable quality.	The report is flawed and doesn't meet the requirements. There may be whole sections missing or poorly covered. There is no coherent professional report structure as required
(30 Marks)						
TOTAL (100)						

Note: Marks will be awarded out of 100 and then reduced to 45 in the calculation of your final grade.