Deficiencies and Omissions

Document metadata: LO4 4.1 EVI 1

Contents:

Main Deficiencies and Potential Remedies

Significant omissions/deficiencies	Comment	Potential Remedies	Affected Tests
Absence of appropriate tools / development	The usage of the React framework made it extremely difficult to compute test coverage via traditional profiling tools due to the interwoven structure of programmatic logic and UI elements of the files.	More research into appropriate structural analysis tools.	T18
Test coverage	The chosen test cases showcased for this portfolio do not cover much of the system. This is due to limited time available, and having the goal of planning and implementing a range of different tests rather then tests with high coverage.	More time budgeted towards A&V processes.	/
Use of inadequate tool	The use of Jest was not appropriate for calculating the test coverage as it could not deal with transpiled code for calculating the test coverage, which was a specific requirement stated in the coursework	Use a different testing framework or different system setup which allows for coverage calculation.	/

	specification. There are Babel plugins for solving this, but I did not have the time to make them work for my system setup.		
Lack of data	The API robustness tests (e.g. the DoS mock attacks) are only somewhat indicative of the system meeting the respective requirement. If we had real-world data of how likely the system is to experience such attacks, and at what scale, this would be invaluable training data and would lead to more sophisticated and useful test cases.	Long-term: add API-use monitoring functionality to the system for recording such kind of data	T4, T12