

CITS3403/5505 Project Marking Sheet.

Name: _____ Student#: _____

Name: _____ Student#: _____

Name: _____ Student#: _____

Name: _____ Student#: _____

Criteria	Excellent	Good	Satisfactory	Inadequate	Comments	Weight
HTML	Valid code, using a wide range of elements, with excellent navigation	Valid code, using a range of suitable elements, and good navigation	Mostly valid code, with adequate navigation	Invalid/incomplete code. Poor navigation		/10
CSS	Valid, maintainable code, using a wide range of custom selectors and classes, reactive to device.	Valid code using a range of custom selectors and classes.	Mostly valid code, using some novel style elements.	Invalid code, or entirely inherited from external packages.		/10
Javascript	Valid, well formatted code, including validation and DOM manipulation/AJAX	Valid, well formatted code, doing client side validation	Some basic client side validation or manipulation, few errors .	Faulty or incomplete code, or entirely from external libraries.		/15
Style	Well considered distinctive style, using coherent images and fonts.	Clear neat style, with some thematic coherency.	Clear readable style, with professional appearance.	Inconsistent or incomplete style.		/10
Content	Excellent and engaging choice of material and mechanism. An application with value.	Well considered choice of material and mechanism. Engaging in the short term.	Effective test mechanism, and usable flow.	Unusable or incomplete application.		/5
Flask Code	Well formatted, commented and organised code, with no errors and nontrivial elements.	Well formatted, commented and organised code, with few errors.	Readable and functional code with few errors.	Unreadable or nonfunctional code, or lacks appropriate referencing.		/10
Data Models	Well considered database schema, good authentication, and maintainable models. Some evidence of DB migrations.	Good database schema with authentication, and effective models.	Functional database schema, with some authentication. Some limitations of data types or redundancies.	Trivial or nonfunctional database, without authentication or session management.		/10
Testing	Comprehensive unit tests, with large test database, and user tests with good coverage.	Well written unit tests covering some module, demonstrated user tests.	Unit tests covering most of some module.	Lacking tests, or trivial tests with low coverage or purpose,		/10
Design	Complex engaging mechanism, fully implemented.	Engaging mechanism, mostly implemented.	Reasonable mechanism and implementation.	Trivial or poorly considered mechanism, incomplete implementation.		/10
Collaboration	Comprehensive Git logs, showing persistent contributions, branching, merging and migrations. Shared contribution to demonstration.	Comprehensive Git logs, showing even contributions and some branching. Shared contribution to demonstration.	Git logs showing contributions from everyone. Everyone present at demonstration.	Missing, uninformative or trivial Git logs. Some people missing at demonstration.		/10
Overall						/100