DALHOUSIE UNIVERSITY
Faculty of Computer Science

CSCI 5708 (Graduate)
Tami Meredith

Mobile Computing Marking Rubric: Assignment 2

Component	Unacceptable: 0-2 *Junk or garbage.*	Minimal: 3-4 *Partially working.*	Good: 5-6 *This is for something OK, but weak.*	Really Good: 7-8 *This is for something OK and effective.*	Exceptional: 9-10 *Work that is better than expected.*
Functionality: *What it does and how well it works.*	It does not execute.	It executes but does nothing.	It executes and does some of what it is supposed to do.	It works effectively and can be used to achieve the task.	It works exactly like it should in all respects. No runtime errors. It has suitable error detection or correction.
Documentation and Code Presentation *Comments, names, code appearance, documents, etc.*	No documentation. Poor code appearance.	There is some internal documentation. That is, the code has decent variable, method, and class names.	The code is effectively commented. The comments add value and explain things you can't see from code reading. Code appearance is good.	User documentation is also available as needed (help, manuals, etc.)	It has outstanding documentation for both programmers and users. Maintaining it is easy because everything is clear and well explained. Designs, test plans, and other documents may exist.
Structure & Design *How clear, structured & maintainable it is. Good use of software engineering principles*.	It has a few classes and methods. It works but feels hacked together. (or) Libraries do most of the work.	It works and has some structure, but not a lot. (or) Libraries do much of the work.	It has abstract data types or some sort of design model. It could be easily extended.	It has abstract data types, it is well structured and has various parts encapsulated. E.g., an input module, a view module, a DB module.	There is a strong design that is documented in some way (e.g., UML, text description). The code is clean and well structured. SOLID, coupling, cohesion and other principles are followed.

1. This assignment will have a mark out of 30. I expect the average score for a component to be somewhere between a 7 and an 8.
2. Structure of the code is important. I want to see some form of code organisation. It should be more than just 2 or 3 classes. Remember: High cohesion, low coupling, encapsulation, abstraction, separation of concerns, etc.
3. Using libraries is OK, but you won't get any marks for anything a library does. Using provided code is expected and required.
4. There must be references for anything looked up or downloaded. Violations of academic integrity will be reported.