field of study Specific Guidelines

Software

project Guidelines

1. Hardware
	* Logical selection of Mac/PC/Raspberry Pi/Arduino/mobile phone/etc. for client application
	* Separate system for hosting the database (preferred)
	* Custom circuits (if a part of the project)
	* Use of University network
2. Software
	* Programming languages beyond ones covered in current classes require advisor’s approval
	* Include a list of non-standard libraries to be used and specific parts used (e.g. ZXing for mobile, Alien RFID SDK for desktop apps, OpenCV – open source machine vision library) and discussion of why the specific libraries and associated libraries were selected
	* Any source code obtained online is limited to libraries only (.jar, .dll, etc.)
	* Include database integration (can be an on-device db for mobile devices) with enforced relationships as appropriate
	* Make use of an external SDK/API (e.g. ZXing bar code decoder library for mobile, Alien RFID reader SDK for desktop)
	* Include some sort of integration with “external” devices (e.g. gyroscope/accelerometer for mobile devices, bar code/RFID reader for desktop)
	* Be modular in design

Project proposal guidelines

1. Project Proposal Documentation
	* Explain the goal you are trying to achieve
	* Process flowchart (high level, conceptual), block diagram(s) of overall system
	* Reason for selection of programming language
	* Include discussion of logic behind selected database engine
	* If custom circuits are to be used, complete and correct schematics are required
	* Project plan / timeline (realistic estimates)
* Prototype development and testing plan
	+ Plan for showing that the desired output is created and meets the goals stated in the project proposal
	+ Utilization of larger data sets

final report guidelines

* Include testing plan
* Include properly documented database scheme
* Allow for user configuration and persistence
* Include block diagram of finished solution
* Include complete software requirements specification as a separate appendix, including documentation for all student developed functions/modules
* Include a user manual
* Results from system tests (including forced error handling, corrupt data, etc.)