MOB Midterm Project - Rubric

Performance					Score	Instructor / TA Comments
App	Exceed (2)	Pass (1)	Doesn't pass (0)	N/A		
App readiness	App is usable and has a clear, easily understandable user interface. App does not exhibit major bugs for common or 'error' use cases.	App meets App Store guidelines. App is usable, though with minor usability quirks that may confuse users. App is stable for common use cases, though not for 'error' use cases.	App is not complete/at all functional. If complete, has serious usability or stability issues that would prevent a user from using the app.			
Execution & scope	App meets all required features, plus some bonus features. Team actively tracked all work items on task tracker.	App meets all or most of the requirements laid out in requirements document, though may have had minor features cut. Team tracked most tasks via a task tracker.	App differs significantly from requirements document. Team did not track tasks.			
Presentation & communication (Optional)	Topic presented with clear description of team's implementation and alternate implementations. Team was able to efficiently communicate and answer questions on the tradeoffs of each approach, how each approach might work, and made an elegant argument for their chosen implementation.	Topic presented to group with a clear description of the team's implementation. Team gave rough overview of other possible implementations. Team was able to answer some questions about their product. Team did not adequately describe why they chose the implementation they did.	Topic presented very minimally. Unclear description of problem to be solved or possible solutions to that problem. Unable to answer audience questions about app imlementation. Did not present alternate implementations to their own, or discuss tradeoffs.	Team did not give presentation		
Technical						
Stability & performance	Manual inspection of code does not have obvious major or minor bugs, and correclty uses all APIs. No significant performance issues. Edge cases caught and errors handled and surfaced correctly. No obvious crashes on manual code inspection.	Code indicates a few small bugs, though for the most part correclty uses Apple's APIs and third party frameworks. Few significant performance issues. Errors caught, though possibly not handled well. Edge cases handled and would not cause app to crash.	Code frequently exhibits bugs or incorrectly uses many APIs/frameworks. Code frequently duplicates behavior already implemented elsewhere (e.g. existing APIs or other parts of student's app). Errors never handled. Edge cases would cause app to crash.			
Style and readability	Code has consistent style throughout codebase in a manner that is consistent with other iOS APIs. Commented well in all unclear areas. Variables and function names clear and consistent.	Code is mostly styled consistently, with a few outliers. A few comments peppered throughout the codebase. Variables and function names make sense and are relatively consistent.	Code is inconsistently styled throughout codebase, or styled in a non-Swift/non-iOS manner. Commented out code abounds. No comments about unclear code. Variable and function names inconsistent and unclear.			
OO and app design	No tightly coupled classes. All classes do one thing and do it well. Elegant use of iOS design patterns. No classes are extremely large. Extensive use of inerhetance/OO concepts. Extremely minimal duplication of code.	Minimal tightly coupled classes. Most classes do one thing, and do one thing well, though a few view controllers are quite large. Moderate use of iOS design patterns. Some use of inheritance/OO concepts. Minimal duplication of code.	Tightly coupled classes about. Most classes have multiple, divergent responsibilities and are quite large. Poor or no use of iOS design patterns. No use of inheritance/OO concepts. Duplicated code abounds.			
					TOTAL	