2022

PS2 Code Judgement Rubric

ONLY CODE JUDGES SHOULD BE USING THIS VERSION OF THE RUBRIC!

Judge (not shared with students): _____
Judged Team (name or number): _____
Date: _____

Instructions: for each requirement, mark the box that most closely aligns to your impression of the solution. Your judgments here should reflect how well the solution satisfies the directions in the problem statement and the team's presentation as well as your own review of the solution's source code. **Aesthetic appeal of a solution is <u>not</u> more important than functionality.** If a solution is not pretty but otherwise meets a requirement, that counts as reasonably satisfying that requirement. Keep your expectations high, teams had **a whole month** to work on this. **DO NOT FEEL OBLIGATED TO AWARD HIGH SCORES!** In the case that a solution attempts to "satisfy" a requirement by making it *look* like it works via the UI/demo code but the required feature is actually non-functional, <u>a score of 0 is appropriate.</u> Also, note that you **do not** have to fill this rubric out in order!

Solution Functionality Review

[total weight: 100]

Requirement(s) to Reference	Score			
	0	1	2	3
	Missing or Completely Non-functional	Attempted But Does Not Satisfy Requirement	Reasonably Satisfies Requirement	Exceeds Requirement Expectations
Change 1	The dashboard view clearly differentiates between the user's answers that have been accepted and those that have not been accepted. **[weight: 2]**			
Change 2	Level 2 users can now upvote comments. Level 5 (given Change 4) users can downvote comments. **[weight: 2]**			
	Users can delete their own comments. **[weight: 2]**			
Change 3	Users receive badges for taking certain actions in the system as described by the problem statement (PS2). **[weight: 12]**			
	Counts representing a user's badges are visible in the app's navigation element as described by the problem statement (PS2).			

	[weight: 10]			
Change 4	Level 4 users can add bounties to questions that do not have an accepted answer, as described by the problem statement (PS2). Bounties are per-app only (questions with bounties in one team's solution may not have the same bounty in another team's solution). **[weight: 22]**			
Change 5	In addition to holding protection, close, and reopen votes, level 7 users can now vote to edit questions and answers as described by the problem statement (PS2). Like the other voting features, edit votes are per-app only (one team's solution will not see the in-progress vote of any other team). **[weight: 22]**			
Change 6 Change 7	Questions can now be "tagged" as described by the problem statement (PS2). **[weight: 5]**			
	Questions can be searched by tag. Tags are per-app only (questions tagged in one team's solution may not have those same tags in another team's solution). **[weight: 5]**			
Change 8	It is obvious in the Buffet view which questions have accepted answers and which don't. **[weight: 10]**			
Change 9	Messages in the Mail view are marked "unread" by default and are marked "read" when opened. This is per-app only ("read" mail on one team's solution may appear "unread" on another team's solution). **[weight: 5]**			
Change 10	You can interact with a "share" button or link, which puts an app-specific link into your clipboard. This link, when navigated to in a new browser tab, leads directly to the exact question/answer that you originally interacted with as described by the problem statement (PS2).			

	[weight: 3]			

Instructions: for each of the five criteria below, mark the box that most closely aligns to your impression of the solution's source code after your review. Does their source code meet your expectations of quality? **If you notice source code that seems strangely out of place, overly complex, extremely nonsensical, suspiciously unformatted or very badly organized syntactically, or you otherwise infer that students copy-pasted/plagiarized part of their solution's source code from the internet or some illegal source, inform the chief judge immediately.** Also, note that you **do not** have to fill this rubric out in order!

Solution Source Code Review

[total weight: 100]

Criterion	Score			
	0	1	2	3
	Does Not Meet Expectations	Partially Meets	Meets Expectations	Exceeds Expectations
Naming Conventions	Variables should be well-named (e.g. **accountSum** is good while **acsu** is bad). There should be few "magic numbers" floating around. The chosen variable naming scheme should be consistent and self-documenting. **[weight 5]**			
Organization	The code should be well-organized, with a consistent logical file/directory layout, encapsulation, and proper separation of concerns. Similar units of code should be grouped together in a logical way. If object-oriented language features are used, they should be used properly (i.e. SOLID). **[weight 30]**			
Maintainability and Extensibility	The solution should be implemented such that a future group of programmers would not have to rewrite large portions of the source code to add new functionality. Source code should be DRY where reasonable. Pay special care to inflexible and hardcoded values. **[weight 25]**			
Readability	How easily readable is the source code? Did you have a hard time going through it? Solution code should be as intuitive and self-documenting as possible. Confusing sections should be documented with comments. There should be few "WTF?!" moments. **[weight 25]**			

Efficiency	Any implementations should not be too inefficient. Does some piece of code waste cycles looping or otherwise doing something unnecessary? Is the application wasteful with system resources? Does it make many more network requests than necessary? **[weight 15]**			

Instructions: please provide any additional comments you have below. Note that your comments are shared directly with the students. Please be thorough, encouraging, and fair.

Positive Comments

Please share any positive comments you have for this team.

Constructive Criticism

Please share any unanswered concerns or comments you have for this team.

Two Questions For This Team

Please share the two most pressing questions you'd like answered about this team's solution.

Shared Thoughts (visible to all judges)

Please add any thoughts or concerns you believe other judges should be made aware of.