2023

PS1 User Experience Judgement 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. Reference the competition problem statement document to get in-depth descriptions of the requirements listed below. Your judgements here should reflect how well the solution satisfies the directions in the problem statement, the aesthetics of the user interface and experience, and the team's presentation. Keep your expectations high, teams had **a whole month** to work on this. **DO NOT FEEL OBLIGATED TO AWARD HIGH SCORES!** Also, note that you **do not** have to fill this rubric out in order! If you have *any questions at all*, please reach out ASAP!

Solution Functionality Review
[total weight: 100]

Requirement(s) to Reference	Score			
	0	1	2	3
	Missing or Completely Non-functional	Attempted But Does Not Meet My Expectations	Meets My Expectations	Exceeds My Expectations
	You **do not have to** login before being able to view the Auth or Profile views. **[weight: 1]**			
	You **must** login before being able to view the full Profile view or the Opportunity view. Must be logged in as an `administrator` to view the Admin view. **[weight: 1]**			
	You **must** login as a `staff` account before being able to view the full Opportunity view. **[weight: 1]**			
Requirement 1 Requirement 6				
	Providing random login credentials *does not* allow you to log into the system successfully. **[weight: 3]**			
	Providing the correct login credentials *does* allow you to log into the system successfully. **[weight: 1]**			
	When you are already logged in, you should not be able to create a new account using the same URL/link used to create accounts when you're not logged in.			

	[weight: 1]		
	You can create a brand new account via a registration feature; you can login successfully using the account you just created. **[weight: 2]**		
Requirement 2	The Home view (the landing page of the app) is pleasant to look at, convincing, and ergonomic. There are no grammatical mistakes or spelling errors. **[weight: 1]**		
Requirement 3	The Profile view (the page displays information about a user) is pleasant to look at, elegant, and ergonomic. It is pretty simple to figure out and use. It has the appropriate components spelled out in the requirement text and provides the functionality spelled out at each bullet point. **[weight: 5]**		
	Users can edit information in their own profile's sections: "About", "Experience", "Education", "Volunteering", and "Skills". **[weight: 3]**		
	Users can change their public profile URL. This change is app-specific and does not apply across solutions. **[weight: 3]**		
	User profiles show Gravatar-based profile pictures and allow uploads of cover photos. This change is app-specific and does not apply across solutions. **[weight: 2]**		
Requirement 4	The Opportunity view (the page that lists links to all the opportunities in the system in chronological order) is pleasant to look at, elegant, and ergonomic. It is pretty simple to figure out and use. It has the appropriate components spelled out in the requirement text and provides the functionality spelled out at each bullet point. **[weight: 5]**		

	A limited version of the Opportunity view is shown to `inner` users while the full version (that allows for the creation, modification, and deleting of opportunities) is shown to `staff` users. `guest` users cannot see this view. **[weight: 2]**			
	Each opportunity has its own unique URL. **[weight: 2]**			
	The Opportunity view supports [Markdown](#). Opportunities are rendered as Markdown and are pleasant to look at. **[weight: 5]**			
Requirement 5	The Admin view (the page allowing `administrators` to view information about the system) is pleasant to look at, elegant, and ergonomic. It is pretty simple to figure out and use. It has the appropriate components spelled out in the requirement text and provides the functionality spelled out at each bullet point. **[weight: 5]**			
	Can force non-`administrator` users to logout. **[weight: 4]**			
	Can impersonate a non-`administrator` user. **[weight: 4]**			
	Can create new accounts through the Admin view. **[weight: 4]**			
Requirement 6	The Auth view (the page that allows you to login and/or register) is pleasant to look at, elegant, and ergonomic. It is pretty simple to figure out and use. It has the appropriate components spelled out in the requirement text and provides the functionality spelled out at each bullet point. **[weight: 5]**			
	You cannot use the same email address or username to register more than one user.			

	That is: usernames and email addresses must be unique. **[weight: 1]**			
	Registration requires a CAPTCHA. Users are restricted from logging in for 1 hour after 3 failed login attempts. Users can see how many attempts they have remaining. **[weight: 2]**			
	When logging in, you can use "remember me" functionality to stay logged in. That is: you can login using "remember me," completely exit the browser, reopen it, and still be logged in afterwards. When you disable "remember me" functionality (i.e. log out, then login again without using "remember me") and you exit and relaunch the browser the same way, you should no longer be logged in. **NOTE: ensure the browser's session restoration "features" are disabled if you're having trouble with this one.** **[weight: 3]**			
Requirement 7	You can use the password recovery feature to regain access to an account without the password. This should be available through a "forgot password" link of some kind when visiting the Auth view. Instructions on how recovery emails are simulated can be found in the team's submission documents. **You MUST be forced to set a new password and should NEVER be shown your old "forgotten" password which would be a dire security failure.** **[weight: 2]**			
Requirement 8	The app has a navigation element with the BDPA logo, app title, and a subset of user data (for authed users) visible in all views. **[weight: 2]**			
Requirement 9	**IMPORTANT:** When logged in and viewing profiles of connected users, it is evident in the UI if the profile is first-order, second-order, or third-order connected. When viewing the profiles of users that are not connected to your user, this is also evident in the UI. **[weight: 10]**			
Requirement 10	**IMPORTANT:** When looking at the profile, opportunity, or through the Admin view, all counts of total views and active viewers/sessions *eventually* (give it 60 seconds) update on the screen automatically when changed in the backend without you having to press a refresh button or take any other actions. You can test this by creating multiple users across multiple tabs that are viewing and interacting with the same opportunities/profiles; you should eventually see the changes from one tab reflected in the other. **[weight: 10]**			

Requirement 12	When viewing lists of profiles, opportunities, etc., all of these results are *paginated* in an aesthetically pleasing way. See the text of the problem statement for more details. **[weight: 1]**		
Requirement 14	The app fails gracefully when errors happen. You don't lose all your progress in the registration form when you press submit with invalid data, for instance. When loading a lot of data, the app does not freeze or present a broken UI but instead shows a spinner icon or some other loading indicator while waiting. You are never waiting too long for something to happen in response to user input. The app never "breaks," never displays internal error messages or warnings anywhere on the page, the backend server hosting the solution never crashes, never any HTTP 500 errors, etc. **[weight: 1]**		
Requirement 15	The app looks good when you view it on a small smartphone-sized screen. **One way to simulate this is by using your mouse to drag the right edge of the browser window to the left until it cannot shrink anymore. This shrunken column represents the size of a phone screen.** Is the app still usable in this state? Is it visually pleasant? **[weight: 2]**		
	The app looks good when you view it on a medium tablet-sized screen. **To simulate this, make your browser window half the size of your screen (try windows key + left arrow). This simulates the size of a tablet screen.** Is the app still usable? Is it visually pleasant? **[weight: 1]**		
	The app looks good when you view it on a normal desktop/laptop screen. **Ensure the browser window is maximized.** Is the app still usable? Is it visually pleasant? **[weight: 2]**		
	Overall, you feel this solution is easy to use. **[weight: 1]**		
	Overall, you feel this solution is beautiful to look at. **[weight: 1]**		

	Overall, you feel this solution functioned well. App functionality that worked fine when the browser window was maximized did not suddenly break when you shrunk the browser to simulate a smartphone or a tablet. **[weight: 1]**			

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.