Buddy App Building Feedback

# Buddy Website

* Signing up for the Buddy service was simple and quick
* Great to see a nuget package for the C# client library
* When you click on the details for an app, the dashboard page is a bit overwhelming
* On the list of users page, why isn’t username one of the fields listed? Seems more useful than the token.
* The “Remember me” on the login page doesn’t seem to work for me in Chrome (didn’t try other browsers)
* Being able to manage users (inactive/delete) via the website interface could be useful. One less thing the app developer would have to build themselves.

# Client Library

* When making service calls, having one callback that handles both success and failure cases is a bit confusing. I was never sure if I should be checking the returned state’s exception property or the return value for null (or false) to know if the call succeeded. I think breaking the error case out into a separate callback would be clearer.
* Several times a call would have a problem but there would be no exception or indication of what when wrong accessible from the C# API. For example, at first I had a bug where I mixed up the latitude and longitude when I made the check-in call and the check-in failed but I had no way of telling why. There was no exception, just a false returned. Also the “API Errors” section of the website did not list any exceptions around check-in calls. I see that the web service has many different error codes that can be returned but I can’t see where these are exposed in the C# library.
* Having some methods be on the BuddyClient class and others on the user class is a bit awkward. It seems that all interaction with the Buddy service should directly through the BuddyClient class.
* After adding a new user profile image, I didn’t see an obvious way to get the URL for the updated image to refresh it in the app. I ended up making another service all to login (since I didn’t see a getUser or refreshUserInfo method) just to get the updated profile image url. It would make more sense for this url to be returned as part of the call to add image.
* I really like how Buddy is scenario based but having all of the scenarios together in the client library is a bit strange. For example, if I am not building a game why would I want to see BuddyClient.Gameboards