Who's writing the tests at ParaBank?

As we saw earlier in this workshop, previous efforts to automation were unsuccessful, in part because ParaBank relied too much on external consultants to do the work for them, without transfer of knowledge and proper training.

This time around, they are looking to be more successful. This is what their approach looks like:

- Developers are responsible for writing, running and maintaining unit tests
- Test automation engineers are responsible for writing, running and maintaining integration and end-to-end tests
- Testers are responsible for exploratory testing

And not without results. It looks like their current approach to test automation, using the distribution of tasks above, yields better results than simply outsourcing the automation of regression tests.

There are still some problems, though:

- After inspection of the integration and end-to-end test code base by a friendly developer, they concluded that there was a lot of overlap in what was covered by unit tests and by the integration tests
- Not all of the testers trust the work that has been done by the test automation engineers, leading to them repeating the tests themselves (and sometimes finding defects that the automation should reasonably have caught)
- Even though they are eager to learn, the test automation engineers are not yet skilled enough in writing good quality integration and end-to-end test code, leading to unnecessary overhead in the form of maintenance time and time needed to add new tests to the code base

To really be able to deliver high quality software at the frequency that the market demands, ParaBank needs to resolve these issues, too. But how?