Test Plan

# Purpose

*Obviously, the purpose is to test your software… But what does “testing” mean in your case? Ensuring data integrity is maintained across synchronized databases? Ensure that UI features function properly and with minimal delays in interactivity? Ensure the installer works? (What does “work” mean?) Are you testing all aspects of your software, only certain parts?*

# Assumptions

*Is there anything your software requires? Something it assumes is working, and that you aren’t responsible for? (Example: Our product relies on the Google Maps API for location data, and as such we don’t verify the accuracy of the information retrieved… we assume it’s correct).*

# Testing procedure

*What steps are you going to take in order to verify <thing that you are testing>?*

* *Manual testing?*
* *Automated scripts or other processes?*
* *Unit tests?*
* *Something else?*

*Regardless of what you are going to do/use, how does that thing work? Explain it here.*

# Tools / Required Resources

*What is necessary to create or execute any of the tests? Are you using any outside resources? Anything to download or install? Do testers (which are often developers too) need to create an account on www.AwesomeTestingToolWeUse.com?*

# Verification

*How do you know the results of the tests? What sort of mechanism do you have to track this? Spreadsheets? Emailing of results? Are the tests quick, with simple screen output of Success/Failure as sufficient indicators?*

*Would someone other than the person who wrote the testing system be able to make any sense of it? Would a new team member be able to interpret results, or use the same tools with a minimal amount of fuss?*