

Alpha and beta testing essay



**ASSIGN
BUSTER**

Introduction | Pre-Production Activities | Lesson Development | Alpha, Beta, and Pilot Testing Project Evaluation | Marketing | Packaging | Dissemination

Alpha Test In software development, your alpha test, will be a test among yourselves (the teams) to confirm that your product works. Originally, the term alpha test meant the first phase of testing in a software development process. The first phase includes unit testing, component testing, and system testing. During this time you will compress files, edit for misspelled words and unclear directions, broken links, and sync audio and video.

You will also test your product on the lowest common denominator machines to make sure download times are acceptable and preloaders work. **Beta Test** In software development, a beta test is the second phase of software testing in which a sampling of the intended audience tries the product out. Beta testing can be considered “ pre-release testing. ” Beta test versions of software are now distributed to curriculum specialists and teachers to give the program a “ real-world” test and partly to provide a preview of the next release.

Pilot Test For the pilot test, we will give your product a “ real-world” test as well as collect data on the use of the product in the classrooms. Here are the steps we will follow for our pilot test.

1. We will recruit test-run professors and teachers who are similar to our intended audience. The important thing here is that the test audience should, as much as possible, be like your “ real” audience so you get the most accurate information.

2. Have the test-run participants use or watch your product. It is not necessary to bring the whole group together at once. It might be better to only have one or two participant test run the product at a time.

3. Observe the test-run participants as they use/watch your product. The important thing here is to try not to interfere. In order to get accurate information, you must not jump in to “ help” as soon as you spot an apparent problem. Of course, if participants really get stuck, you do want to work with them so they can continue to test your product.

a. Make notes. Your notes should include information about where any problems occurred, under what circumstances, and how the person attempted to resolve or actually did resolve the problem. You should include any participant reactions, both positive and negative, which you observe. Your notes should include information such as “ Screen #10 - both participants clicked on the big picture of the car instead of the first small picture” or “ the right arrow button on screen #3 sends user to screen #4 instead of screen #7. ”

b. Ask questions. Your questions should help clarify why people are doing what they’re doing (i. . , When you got to the screen with the one big and several small pictures of cars, why did you click on the big picture first?). Your questions should also help you make changes (i. e. , What would you suggest we do to make this screen, page, frame, etc. less confusing?).

4. Have the test-run participants make notes as they use/watch your product. This is a good way for the test-run participants to capture things as they happen. You will likely get confirmation of problems you observed as well as some on-the-spot thinking which you cannot observe very easily. This

is also a useful time to collect information if you are unable to directly observe the test-run participants.

5. Have the test-run participants complete a survey. This is a more systematic way to collect the information you are after. All test-run participants answer the same questions, and you can quickly see any trends that develop. Like the test-run participant notes, a survey is also a good way to get information if you are unable to observe the participants yourself.

6. Conduct interviews/focus group after the test-run.