To avoid creating software customers don't like, we must constantly focus on two important questions:
1) Are we building the right product?
2) Are we building the product right?

1) Validation
    a. Are we building the right product?
    b. Does the product we're building meet the customer's needs?
    c. Validation activities
        i. Accurately and completely capturing customer requirements
        ii. Prototyping
        iii. Delivering frequent builds that users can test drive
        iv. Usability testing
        v. System testing
        vi. Alpha/Beta testing

2) Verification
    a. Are we building the product right? (If we're building the right product, and we also build it right, then we succeed.)
    b. Validation ensures that customer requirements are met. Verification ensures that the system is built according to spec.
    c. Verify that the product of each development phase is complete and consistent with respect to the product of the previous phase. (This can be done even if you're building the wrong product.)
        i. Does the product defined in the functional spec satisfy all requirements in the PRD?
        ii. Does the design cover everything in the functional spec?
        iii. Do our unit tests cover everything in the design?
        iv. Does the test plan test every feature described in the functional spec?
        v. Etc.
    d. Every aspect of downstream work products should be **traceable** back to the customer requirements. Every requirement should be **traceable** to the downstream work products that realize it.
        i. Does the functional spec contain features the customer didn't ask for?
        ii. Are developers secretly adding cool features the customer didn't ask for?
        iii. Are we testing things that don't matter to the customer?