

1. Em minhas pesquisas só encontrei os níveis de testes já relatados pela professora, que são:
 - a. Testes de Unidade: Uma responsabilidade do desenvolvedor; serve para testar o comportamento de cada função, evitando erros, exceções não tratadas, comportamento inesperado
 - b. Testes de Integração: Uma responsabilidade do desenvolvedor, serve para testar como duas ou mais funções se comportam quando executadas no mesmo fluxo, e corrigir erros que possam quebrar o programa.
 - c. Teste de sistema: Uma responsabilidade da equipe de testes; serve para ver se o sistema se comporta como esperado, seguindo as regras de negócio, tratando erros que podem aparecer.
 - d. Teste de aceitação: Feito pelo usuário, serve para dizer se o sistema chegou a sua versão final e está agradável para o usuário, e o teste da ponta e diz respeito mais a usabilidade do que tratamento de erros
2. Sobre ferramentas de testes manuais não encontrei nenhuma específica, a única que me pareceu próxima foi uma chamada teste de 5 segundos, que mostra uma parte do software por 5 segundos e pergunta a respeito dele, a fim de descobrir se identidade visual corresponde ao seu objetivo
 - a. O motivo de usar ferramentas de software é porque elas oferecem confiabilidade, desde que configuradas corretamente, e também oferecem um ganho de produtividade, já que fazem mais testes em menos tempo.
 - b. Sobre ferramentas: Selenium, Cypress, Mocha, PHPUnit, JSUnit, já ouvi falar ou mexi com essas ferramentas.