

### LISTA 03

1. Systematycznie uzupełniaj kod aplikacji.
2. Omów testy jednostkowe. Wykonaj testy jednostkowe dla kilku klas, można użyć narzędzi zob. <https://www.softwaretestinghelp.com/unit-testing-tools/> Napisz metody testowe, pokaż efekt działania tych metod.
3. Przygotuj testy akceptacyjne, zob. np. <https://testerzy.pl/baza-wiedzy/artykuly/rola-testowania-akceptacyjnego-uat>. Możesz wykorzystać różne narzędzia np. FitNesse (zob. wykład – podane różne narzędzia). Wykonaj testy akceptacyjne dla kilku klas. Pokaż efekt działania testów.
4. Przygotuj i wykonaj testy funkcjonalne, możesz użyć narzędzia np. Selenium zob. <https://www.selenium.dev/selenium-ide/docs/en/introduction/getting-started> <https://www.selenium.dev/documentation/overview/> ((możesz użyć innego narzędzia). Pokaż efekt działania testów.
5. Wykonaj testy manualne, przedstaw efekt działania tworzonej aplikacji.
6. Opisz wyniki testów, wyciągnij wnioski.

**Do oceny oddajesz:** uzupełniony plik \*.pdf (z LISTY 01, LISTY 02) zawierający dodatkowo powyższe elementy, w którym powinny znaleźć się też widoki ekranu potwierdzające implementację aplikacji (uzupełnianie kodu źródłowego aplikacji).

**Dodatkowo do oceny oddajesz:** wykonaną aplikację, zgodnie z ustalonym tematem, która będzie przetestowana przez prowadzącego zajęcia pod kątem zgodności z postawionymi na początku wymaganiami.

**Termin:** zgodnie z kalendarzem umieszczonym w osobnym pliku.