## Testy

Testowany komponent	Kroki testowania	Oczekiwany wynik	Potwierdzenie rezultatów
Pobieranie danych ze źródła 1 (Spotify)	1. Wykonanie aplikacji load_spotify.py	Uzyskanie plików albums.csv, bands.csv, genres.csv	C:\Users\hp\Documents\github\BD3-AO3-main\Spotify Download>python load_spotify.py Downloading BIS Done! Downloading One Direction (Band) Done! Downloading KEV (Band) Done! Downloading SEVENTEEN (Band) Donel Downloading SEVENTEEN (Band) Donel Downloading SEVENTEEN (Band) Donel Downloading STray Kids (Band) Donel Downloading GOT7 Donel Downloading Monsta X (Band) Donel Downloading Monsta X (Band) Donel Downloading Monsta X (Band) Donel Downloading ATEEZ (Band) Donel Lonel Downloading ATEEZ (Band) Donel C:\Users\hp\Documents\github\BD3-AO3-main\Spotify Download>_
Pobieranie danych ze źródła 2 (ao3)	Wykonanie aplikacji     ao3download.py	Uzyskanie plików ao3_band_tag_clean.cs	Rezultaty repozytorium w notatniku BD3-AO3/AO3 download/ao3.ipynb
		ao3_band_fic_clean.csv	
Tworzenie struktury folderów w hdfs	Wykonanie sekwencji komend w terminalu     Sprawdzenie     struktury folderów	Foldery <b>/ao3</b> i <b>/spotify</b> w hdfs	Wylistowanie struktury folderów w hdfs przed i po wykonaniu komend  ***Toward of the struktury folderów w hdfs przed i po wykonaniu komend  ***Toward of the struktury folderów w hdfs przed i po wykonaniu komend  **Toward of the struktury folderów w hdfs przed i po wykonaniu komend i po
Konwersja danych do formatów avro i parquet i zapisanie ich w hdfs	1. Uruchomienie flow NiFi 2. Wylistowanie plików w folderach <b>spotify</b> i <b>ao3</b> w hdfs	Pliki ao3_band_tag_metadata _clean.avro, ao3_band_tag_metadata _clean.parquet, ao3_band_fic_metadata _clean.avro i ao3_band_fic_metadata _clean.parquet w folderach spotify i ao3 w hdfs	Wylistowanie plików w hdfs przed wykonaniem flow NiFi  Wykonaniem flow NiFi  Wykonaniem flow NiFi  Wykonaniem place w przed wykonaniem w przed wykonaniem place w przed w prze



