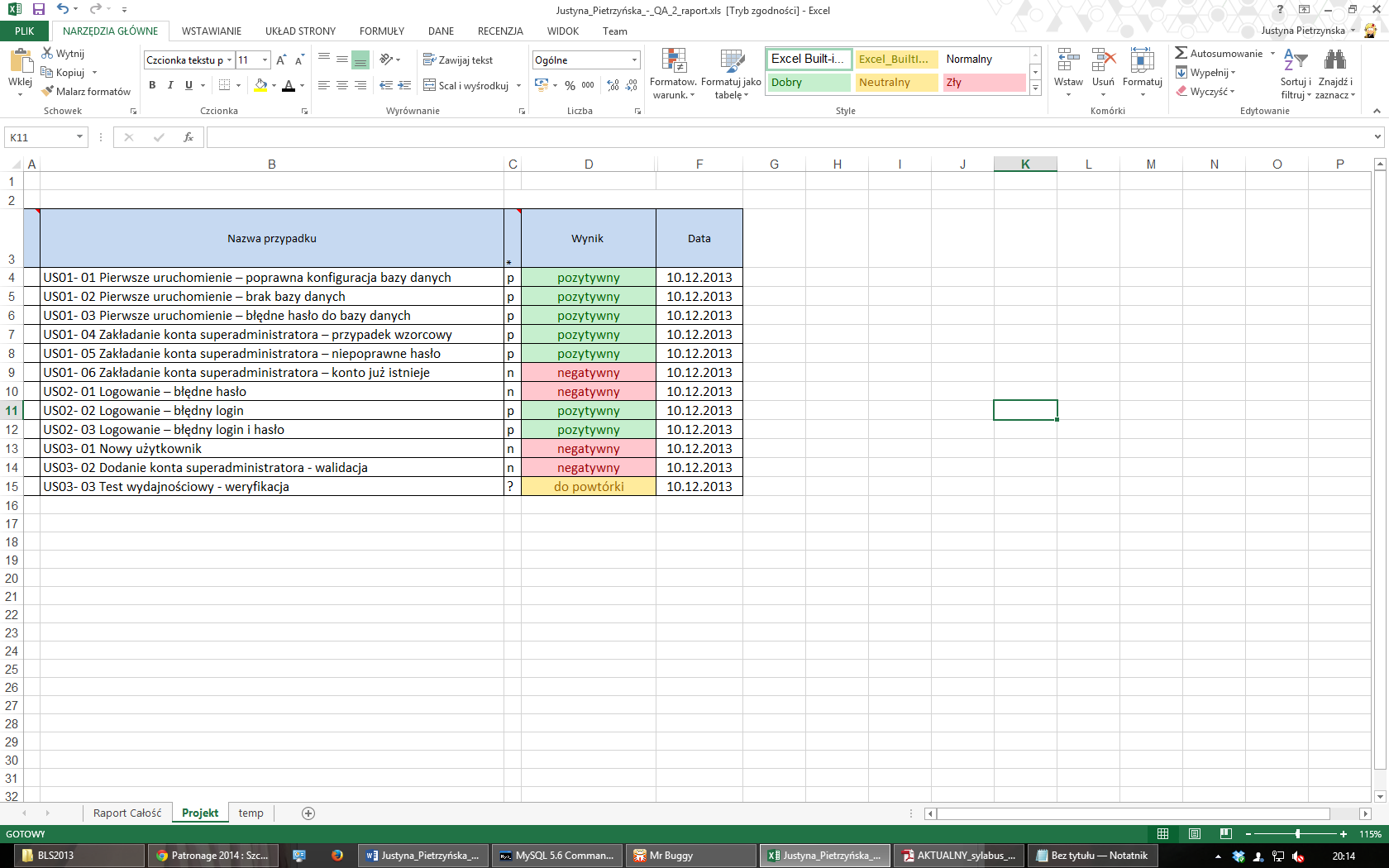
# Testy



Całość raportu dostępna jest w załączniku Justyna\_Pietrzyńska\_-\_QA\_2\_raport.xls

# Raportowanie błędów

## Duplikowanie nazwy konta

| Tytuł | Duplikowanie nazwy konta – US01-06 |
| --- | --- |
| Warunki początkowe | Komputer z zainstalowanym systemem Windows oraz środowiskiem MySQL 5.6., utworzona baza *mr\_buggy*, w bazie istnieje użytkownik *justek*, uruchomiona aplikacja Mr Buggy. |
| Kroki do wykonania | * Uruchomienie aplikacji * Zalogowanie do bazy danych mr\_buggy * Utworzenie konta użytkownika o nazwie *justek* |
| Efekt | Zostaje utworzone kolejne konto o tej samej nazwie. |
| Oczekiwany rezultat | Według specyfikacji (str 10, punkt 7.8) nazwa użytkownika musi być unikalna. |
| Rezultat wykonania testu | Negatywny. |

## Trzykrotne błędne logowanie

| Tytuł | Trzykrotne błędne logowanie |
| --- | --- |
| Warunki początkowe | Komputer z zainstalowanym systemem Windows oraz środowiskiem MySQL 5.6., utworzona baza *mr\_buggy*, uruchomiony serwer MySQL, uruchomiona aplikacja Mr Buggy, utworzone konto o nazwie *iustek* i loginie *blstream2013!* |
| Kroki do wykonania | * Trzykrotna próba logowania błędnym hasłem (*justek/blstream2012?)* * Zalogowanie poprawnym hasłem |
| Efekt | Użytkownik zostaje zalogowany. |
| Oczekiwany rezultat | Według specyfikacji (strona 6, punkt 6) Trzykrotne podanie błędnego hasła dla danego loginu/adresu email powoduje zablokowanie konta w systemie. Kolejne próby zalogowania się powodują wyświetlenie komunikatu (jak wyżej) dla zablokowanego konta i zalogowanie się nie jest możliwe. |
| Rezultat wykonania testu | Negatywny. |

## Polskie znaki podczas tworzenia użytkownika

| Tytuł | Polskie znaki podczas tworzenia użytkownika |
| --- | --- |
| Warunki początkowe | Komputer z zainstalowanym systemem Windows oraz środowiskiem MySQL 5.6., utworzona baza *mr\_buggy*, uruchomiony serwer MySQL, uruchomiona aplikacja Mr Buggy, widoczny formularz dodawania konta. |
| Kroki do wykonania | * Wpisanie w odpowiednie pola danych wejściowych, nazwa użytkownika: żółćgęśląjaźń * Kliknięcie przycisku OK |
| Efekt | Użytkownik nie zostaje utworzony. |
| Oczekiwany rezultat | Pole powinno przejść walidację poprawnie. |
| Rezultat wykonania testu | Negatywny. |

## Polskie znaki podczas tworzenia użytkownika

| Tytuł | Polskie znaki podczas tworzenia użytkownika |
| --- | --- |
| Warunki początkowe | Komputer z zainstalowanym systemem Windows oraz środowiskiem MySQL 5.6., utworzona baza *mr\_buggy*, uruchomiony serwer MySQL, uruchomiona aplikacja Mr Buggy, widoczny formularz dodawania konta. |
| Kroki do wykonania | * Wpisanie w odpowiednie pola danych wejściowych, nazwa użytkownika: żółćgęśląjaźń * Kliknięcie przycisku OK |
| Efekt | Użytkownik nie zostaje utworzony. |
| Oczekiwany rezultat | Pole powinno przejść walidację poprawnie. |
| Rezultat wykonania testu | Negatywny. |

## Dodawanie konta superadministratora

| Tytuł | Dodawanie konta superadministratora |
| --- | --- |
| Warunki początkowe | Komputer z zainstalowanym systemem Windows oraz środowiskiem MySQL 5.6., utworzona baza *mr\_buggy*, uruchomiony serwer MySQL. |
| Kroki do wykonania | * Uruchomienie aplikacji |
| Efekt | Pojawia się formularz dodawania konta superadministratora. |
| Oczekiwany rezultat | Konto superadministratora powinno być tworzone przy pierwszym uruchomieniu, nie przy każdym uruchamianiu aplikacji (specyfikacja str. 7, punkt 7.1). |
| Rezultat wykonania testu | Negatywny. |

# Podsumowanie

### Wyjaśnij co oznaczają pojęcia Gray BoxTesting oraz Glass BoxTesting.

Jest to połączenie white-box testing i black-box testing, w której testerzy znają specyfikację a jednocześnie znają budowę aplikacji, celem takiego testowania jest poszukiwanie ewentualnych wad na skutek nieprawidłowego użytkowania lub niewłaściwe struktury aplikacji.

### It’snot a bug, it’sa feature. Spróbujcie w 1 zdaniu uzasadnić różnicę między błędem, a funkcjonalnością.

### Funkcjonalność jest zamierzonym efektem działania aplikacji, błąd – niezamierzonym.

### Patrząc na dotychczasowe czynności oceńcie czy można w 100% przetestować oprogramowanie?

Nie ma możliwości przetestowania oprogramowania w 100%, należy dążyć do jak największego pokrycia testami możliwych instrukcji oraz decyzji. Wiele projektów przyjmuje metrykę zakładającą, że 80% kodu powinno być pokryte testami.

### Kiedy można zakończyć testowanie?

W momencie, kiedy pokrycie kodu jest wysokie, krzywa wykrywania błędów poniżej założonego wcześniej progu a ryzyko wystąpienia błędów jest minimalne i koszty z nim związane są mniejsze niż dalsze testowanie.