TASK TO DO NA 21.10.2016:

1. Przechodzimy z C# na PHP, pamiętajcie więc że wszystko odbywa sie w środowisku przeglądarki internetowej przez co trzeba poprawić, wprowadzić korekty wszystko co do tej pory było zrobione.

**2. Inżynier wymagań**

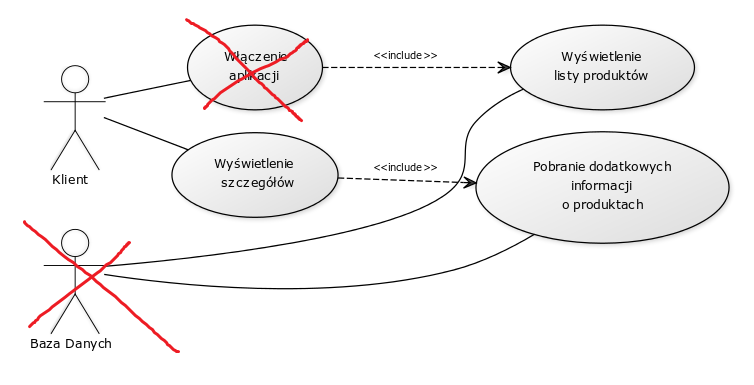
Poprawić napisany do tej pory tekst, info ogólne, użyte technologie itd.

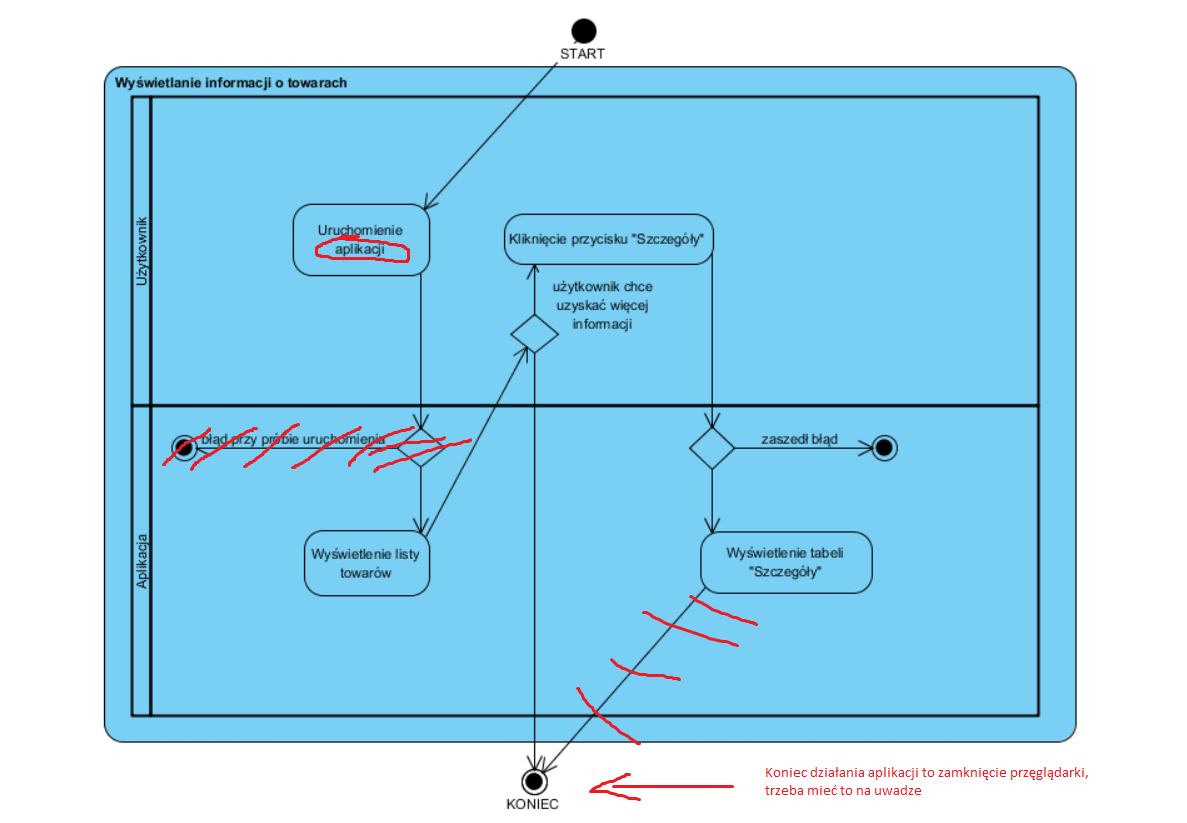
Funkcjonalności:

* Wyświetlanie produktów
* Szczegóły produktów
* Edycja szczegółów
* Dodawanie produktu

**3. Dla architekta:**

* Poprawić 1 use case diagram,



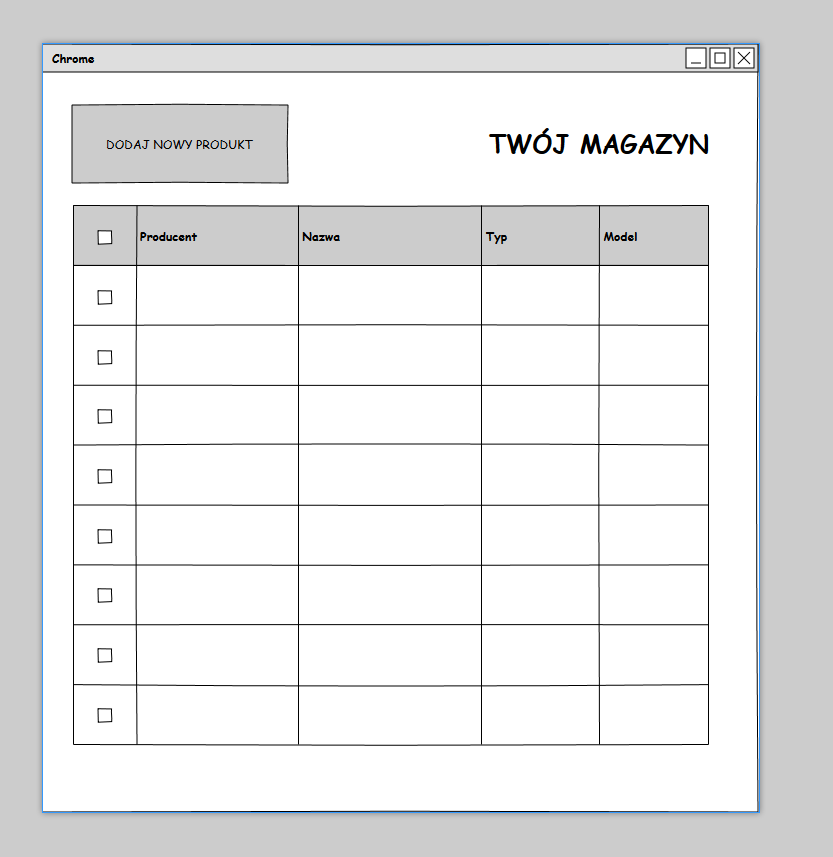
* dopisac do tego funkcjonalnośc: edycja, dodawanie nowego
* poprawka dla diagramu aktywności
* stworzyć diagram aktywności dla edycji i dodawania produktu.

**4. Dla deweloperów**

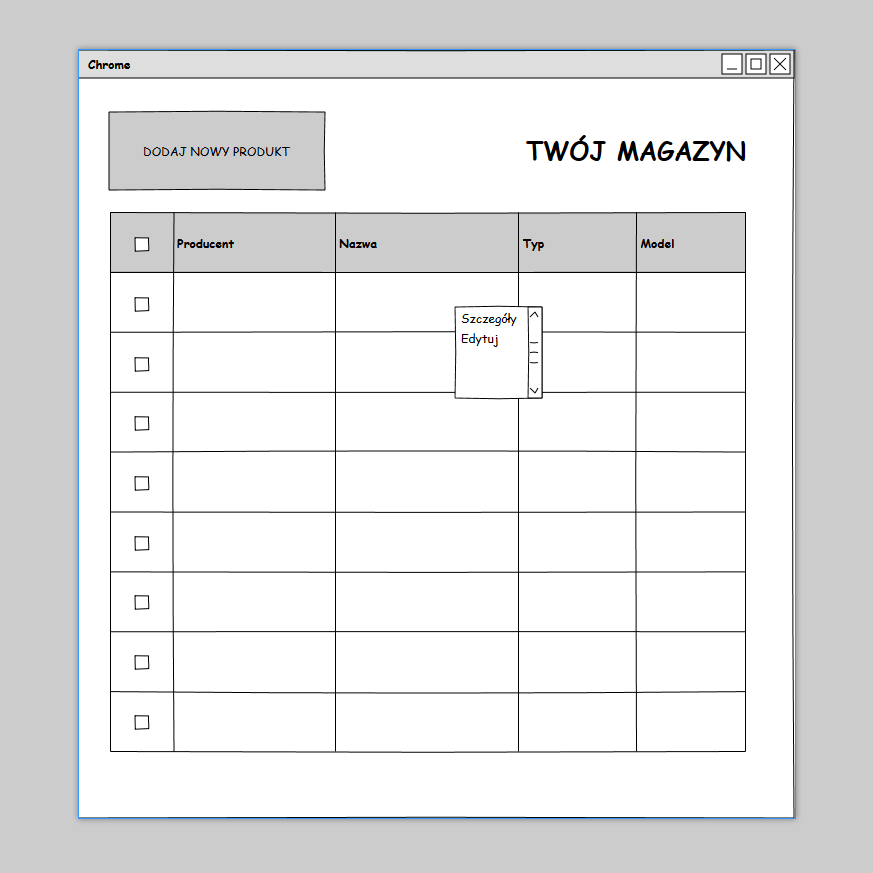
* przepisać na php to co do tej pory
* dopisać funkcjonalność edycji i szczegółów
* **STOWRZYĆ DIAGRAM SEKWENCJI DLA WSZYTKICH 3 FUNKCJONALNOŚCI**
* wygenerować diagram obiektów i klas ( można to zrobić w visual studio prawym przyciskiem myszy na plik źródłowy)

przykłady tych diagramów są we wzorze który wrzuciłem na samym początku (plik: przykładowa\_dokumentacja\_techniczna)

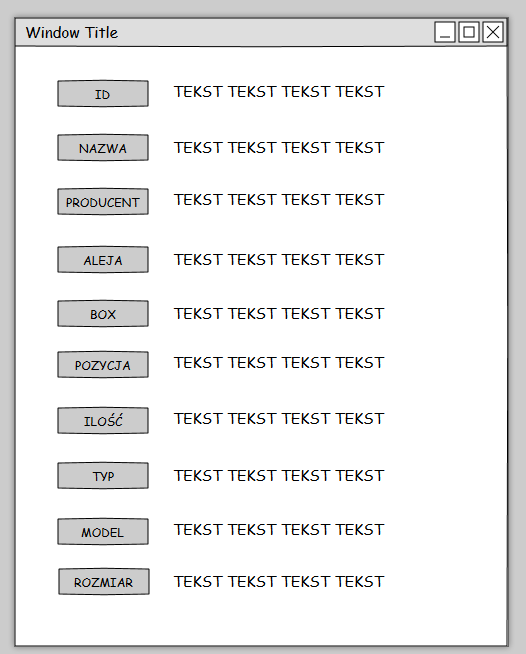
* Jak to ma wyglądać (zrobiłem poglądowe rysunki by wam nie narzucać od górnie, jak wam wygodniej i według gustu )



KOMÓRKA NIE MOŻE BYĆ ZAZNACZALNA, JAK COŚ ZAZNACZA SIĘ CAŁY WIERSZ



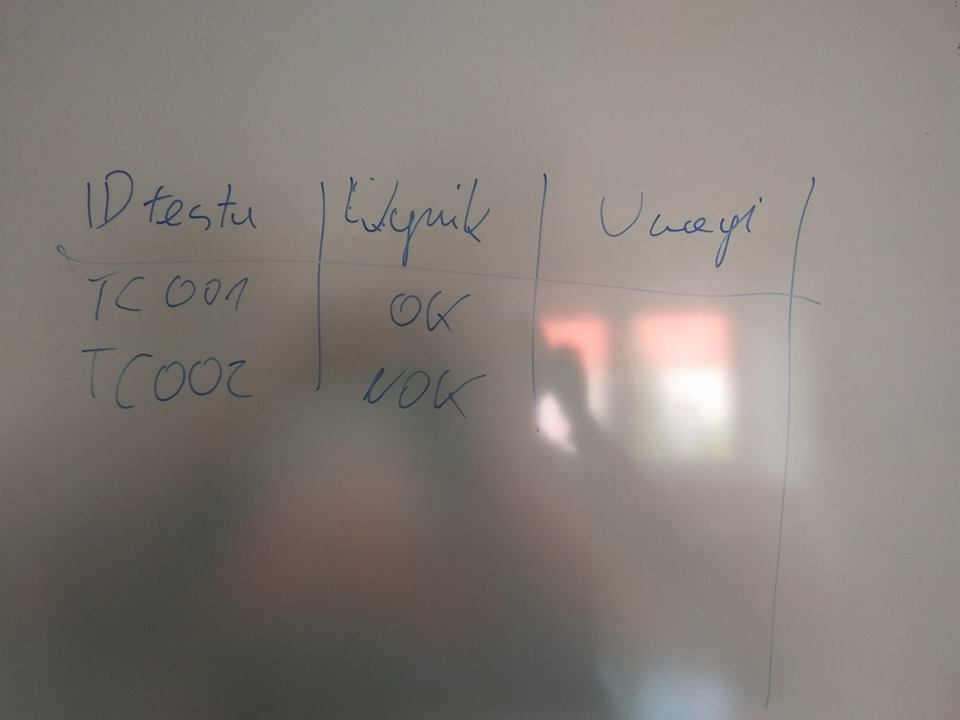
NACISKAMY PRAWY PRZYCISK MYSZY I POJAWIA SIE MENU ROZWIJANE



SZCZEGÓŁY MAJĄ BYĆ NIE EDYTOWALNYMI POLAMI, JEDYNIE PRZEZ TAMTĄ OPCJE W MENU, POWIEDZIAL ŻE LEPIEJ BY BYŁ TO TEKST A NIE TAK JAK W PIERWSZEJ WERSJI POLA IN PUT, SKORO NIE EDYTOWALNE POLA ( ale jak wam wygodniej i ładniej)

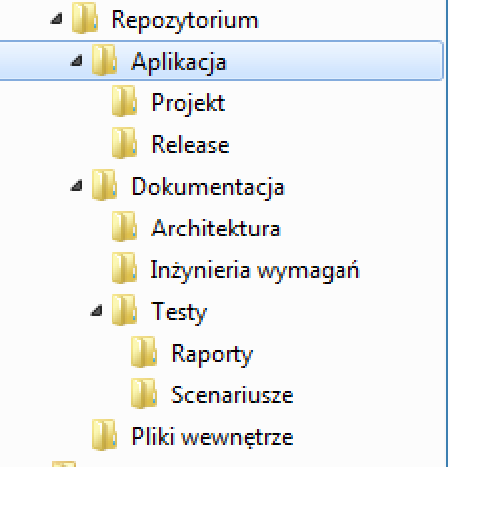
**5. Tester**

* dopisać id do scenariuszów testowych
* poprawić scenariusze do poprawek naniesionych po zmianach innych osób (małe korekty w funkcjonalnościach i architekturze wniosą zmiany w scenariuszach)
* dopisać scenariusze testowe dla nowych funkcjonalności
* wygenerować na koniec raport ( dwa raporty, pierwszy do pierwszego scenariuszu, drugi do drugiego ):



**6. Integrator**

* poprawić merge w githubie, usystematyzować je jakoś, ogarnąć to po prostu.
* stworzyć strukturę naszego repozytorium:



W folderze repozytorium > aplikacja> projekt ma być aktualny projekt naszej apki.

W folderze repozytorium > aplikacja> REELEASE ma być sama apka.

W folderze Dokumentacja chyba wszystko jasne każdy ma swój katalog, tester ma dwa w jednym trzyma scenariusze w drugim raporty

Pliki wewnętrzne to nasze pliki które wrzucamy tam bo coś trzeba, tak jak TEN DOKUMENT.