



Wisielec Schopenhauera

(tytuł roboczy)

Maciej Janiszewski

19.10.2016

Streszczenie

Celem projektu jest napisanie nowoczesnej wariacji na temat gry w wisielca z oprawą graficzną związaną z tematami nihilistycznymi oraz możliwością gry wieloosobowej.

Wprowadzenie

Wisielce jest grą której nie trzeba nikomu przedstawiać. Prosta mechanika pozwala nawet niedoświadczonym graczom błyskawicznie zacząć rozgrywkę. Jest to prawdopodobnie trzeci najczęściej wybierany, najbardziej dostępny zabójca czasu obok kółka i krzyżyka oraz gry w państwa-miasta/inteligencję. W dobie wysokiej popularności urządzeń mobilnych typu smartfon/tablet, koncepcja gry w wisielca nadal nie doczekała się godnego odpowiednika w postaci aplikacji mobilnej bijącej rekordy popularności (z drugiej strony, scrabble miały więcej szczęścia).



Cel

Aby wprowadzić wisielca w XXI wiek, należy posłużyć się obecnie stosowanymi praktykami na rynku aplikacji/gier mobilnych. Gracz musi przede wszystkim chcieć wracać do tej gry. Jednym ze sposobów, aby zatrzymać gracza, jest danie mu poczucia celu – na przykład w formie odblokowywanych osiągnięć i trybów (może chcieć zobaczyć, co będzie dalej), albo rankingów (każdy chce być najlepszy). Obecne produkty znajdujące się w sklepie Google Play nie są zbyt innowacyjne. Przeważają dwie odmiany, jedną jest klasyczny wisielce bez żadnej szczególnej oprawy graficznej, zapewne napisany na podstawie prostego tutoriala i wrzucony do sklepu. Drugą odmianą jest klasyczny wisielce wzbogacony o elementy typowej aplikacji mobilnej (podział na etapy i poziomy), reklamy, mikropłatności oraz miłą dla oka (jednak nie wyróżniającą się) oprawą graficzną. Mój produkt będzie bardziej innowacyjny i dostępny na wielu platformach.

Wymagania

Platforma docelowa

Po stronie klienta wymagana jest dowolna nowoczesna przeglądarka ze wsparciem dla technologii HTML5 (co najmniej Internet Explorer 11). Należy uwzględnić, że wymagania dla infrastruktury serwerowej rosną wraz ze zwiększeniem się ilości graczy. Aplikacja będzie działać na dedykowanym serwerze, za serwowanie plików statycznych docelowo będzie odpowiadać nginx, zaś hostować samą grę będzie Unicorn. Za bazę danych posłuży PostgreSQL.

Funkcjonalności produktu

Produkt będzie spełniał następujące założenia:

1. Możliwość gry wieloosobowej, jak i samemu.
2. Generowanie haseł na podstawie danych z serwisów Wikisłownik i Wikicytaty.
3. Wiele trybów (rozgrywka turowa, w czasie rzeczywistym itd.)
4. Podstawa gry - klasyczna mechanika gry w wisielca (jeden gracz zadaje hasło, drugi je zgaduje poprzez podawanie różnych liter, jest dozwolona ilość pomyłek).
5. Zaimplementowanie różnych wariacji od oryginalnej koncepcji (do przemyślenia w trakcie projektowania samej gry i do przetestowania na rzeczywistych odbiorcach), np. tryb samobójczy w którym gracz dostaje długie hasło i musi wybrać litery, których w nim nie ma. Warto pomyśleć o zaimplementowaniu tego w formie modyfikatorów (jak np. w osu!).

6. Obsługa kont użytkowników, logowania przez portale społecznościowe, udostępniania swoich wyników przez fb, system osiągnięć.
7. System rankingów dla najlepszych graczy.
8. W grę powinno dać się grać bez zakładania konta.
9. Lobby (lista gier i czat), możliwość dołączenia do istniejącej gry poprzez zaproszenie lub link, podobnie jak w serwisie kurnik.pl
10. Oprawa graficzna zaprojektowana wokół następujących pojęć kluczowych: schopenhauer, wisielec, Halloween, trupie czaszki, cmentarze, trumny, zombie, nihilizm, szubienica, bezsens istnienia
11. Aplikacja musi być zgodna z założeniami Responsive Web design.

Harmonogram

Projekt podzielony zostanie na 4 etapy,

Etap 1 – deadline 10 listopada 2016

- komunikacja klient<->serwer, działająca gra (bez logowania, łączenie się po ID, zakładanie nowej),
- makietę interfejsu,
- działający prototyp,
- lista gier

Etap 2 – deadline 1 grudnia 2016

- autentykacja (logowanie, rejestracja itd.),
- widok profilu, zaproszenia,
- system rankingowy,

Etap 3 – deadline 1 stycznia 2016

- czat,
- modyfikatory,
- system rankingowy,

Etap 4 – deadline 12 stycznia 2016

- oprawa graficzna,
- poprawki zgodności z RWD



Technologie

Do stworzenia produktu posłużę się frameworkiem do tworzenia aplikacji webowych o nazwie Django. Wykorzystuje on język skryptowy Python do obsługi backendu. Komunikacja klient <-> serwer zostanie zrealizowana za pomocą protokołu WebSocket. Produkt zostanie zrealizowany w formie aplikacji webowej zgodnie z założeniami Responsive Web Design (interfejs aplikacji będzie dostosowywał się do urządzenia na którym będzie ona wyświetlana) z użyciem języków JavaScript, CSS oraz HTML.