

Instytut Informatyki Uniwersytetu Wrocławskiego

Kamil Galik, Wiktoria Kuna

D&Direct

Koncepcja wykonania systemu

Wrocław, 24 października 2022
Wersja 1.0

Data	Nr Wersji	Opis	Autor
22.11.2022	0.1	Utworzenie dokumentu	Kamil Galik
23.11.2022	1.0	Korekta dokumentu	Wiktoria Kuna

Spis treści

1. Wprowadzenie
2. Scenariusze przypadków użycia
 - 2.1. Schematy interakcji na podstawie scenariuszy użycia
 - 2.2. Projekt widoków na podstawie scenariuszy użycia
3. Schemat bazy danych
4. Model konceptualny rzeczywistości
5. Projekt architektury systemu
6. Identyfikacja zagrożeń
7. Porównanie z tablicą koncepcyjną i specyfikacją wymagań

1. Wprowadzenie

W tym dokumencie opisana jest koncepcja realizacji systemu. Jej głównymi elementami są: scenariusze przypadków użycia systemu, projekt architektury systemu i schemat bazy danych. Ponadto znajduje się tu model conceptualny rzeczywistości, w oparciu o który projektowany jest nasz system.

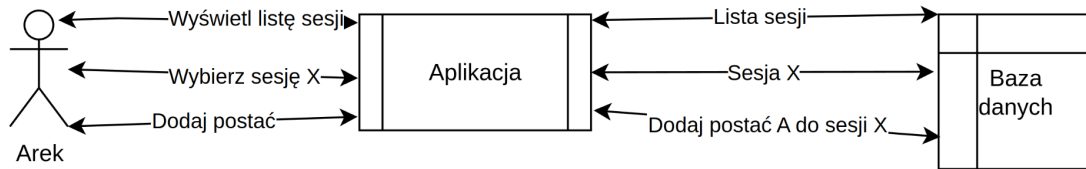
Równie ważnym elementem jest identyfikacja zagrożeń dla powstania systemu.

Dokument wieńczy analiza różnic bieżącej koncepcji z uprzednio przygotowaną tablicą koncepcyjną i specyfikacją wymagań,

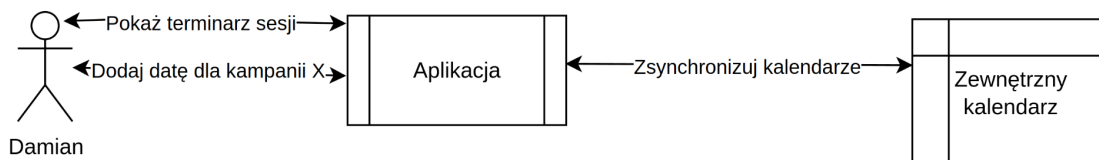
2. Scenariusze przypadków użycia

- Arkadiusz musi dodać bohatera do jednej ze swoich sesji, ponieważ jego kolega powiedział, że chce dołączyć. Otwiera więc aplikację i zauważa 3 kafelki: terminarz sesji, narzędzia RPG i spis aktywnych sesji. Naciska w kafelek "spis aktywnych sesji" gdzie widzi kolejne opcje: spis bohaterów, dziennik sesji i narzędzie do map. Arek wybiera "spis bohaterów" i znajduje nad pierwszą pozycją na liście przycisk ze znakiem plus, który odpowiada za dodanie bohatera. Pojawia się formularz, w którym należy wypełnić dane postaci oraz przycisk z ikoną kostki do gry. Arek naciska przycisk kostki, by wylosować cechy postaci i zapisuje zmiany. Następuje przesłanie zapytania do bazy danych, aby utworzyć nowy wpis w tabeli z bohaterami.
- Damian umówił się z kolegami na sesję, ale nie dopisał jej jeszcze do terminarza. Otwiera więc aplikację i wybiera opcję "terminarz sesji". Znajduje tam kalendarz, który ma wpisane w konkretne dni informacje o planowanych sesjach. Konkretnie w jakich godzinach ma się odbyć i której kampanii dotyczy. Damian wybiera ustalony dzień i pojawia się okno z listą wyboru sesji oraz możliwością wpisania zakresu godzinowego. Po dokonaniu wyboru informacja zostaje zapisana w danych aplikacji i następuje synchronizacja z zewnętrznym kalendarzem, jeżeli taki został uprzednio wybrany przez Damiana.
- Michalina jest w trakcie prowadzenia sesji. Potrzebuje wykonać serię rzutów kością, aby obliczyć obrażenia zadane w walce. W tym celu korzysta z dostępnej opcji "narzędzia RPG". Po wybraniu tej opcji widzi listę dostępnych narzędzi. Wybiera ikonę z napisem "symulator rzutów", a następnie przy pomocy edytora tworzy listę rzutów kostką. Przedstawia się to jako lista z przyciskiem do dodawania kostek, przyciskiem do wykonania rzutów z listy oraz przyciskiem do wyczyszczenia listy. Przy dodawaniu kostki pojawia się pole wyboru jakiego typu jest to kostka (uwzględnione są tylko rodzaje używane przy D&D) oraz przycisk "dodaj". Michalina może swobodnie zmieniać kolejność kości przy pomocy przeciągania ich oraz przy każdej pozycji jest przycisk z ikoną minusa do usuwania pozycji z listy. Po ułożeniu satysfakcjonującej sekwencji, wykonuje rzut wszystkimi kośćmi i obok każdej z pozycji pojawia się wynik rzutu. Michalina powtarza rzuty parokrotnie, po czym wraca do prowadzenia gry.

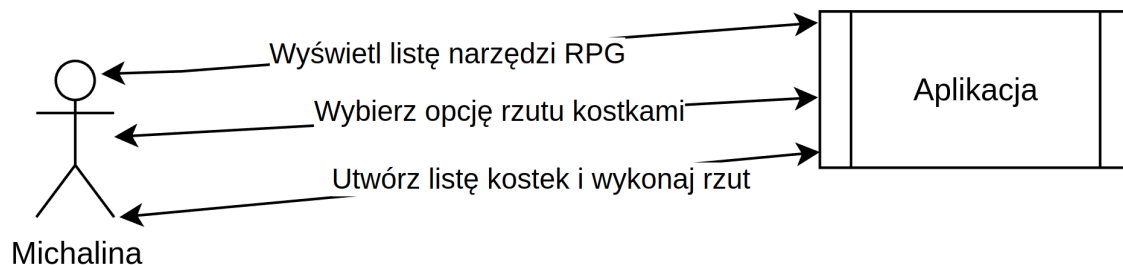
2.1. Schematy interakcji na podstawie scenariuszy użycia



Przykład 1. Arek dodaje postać

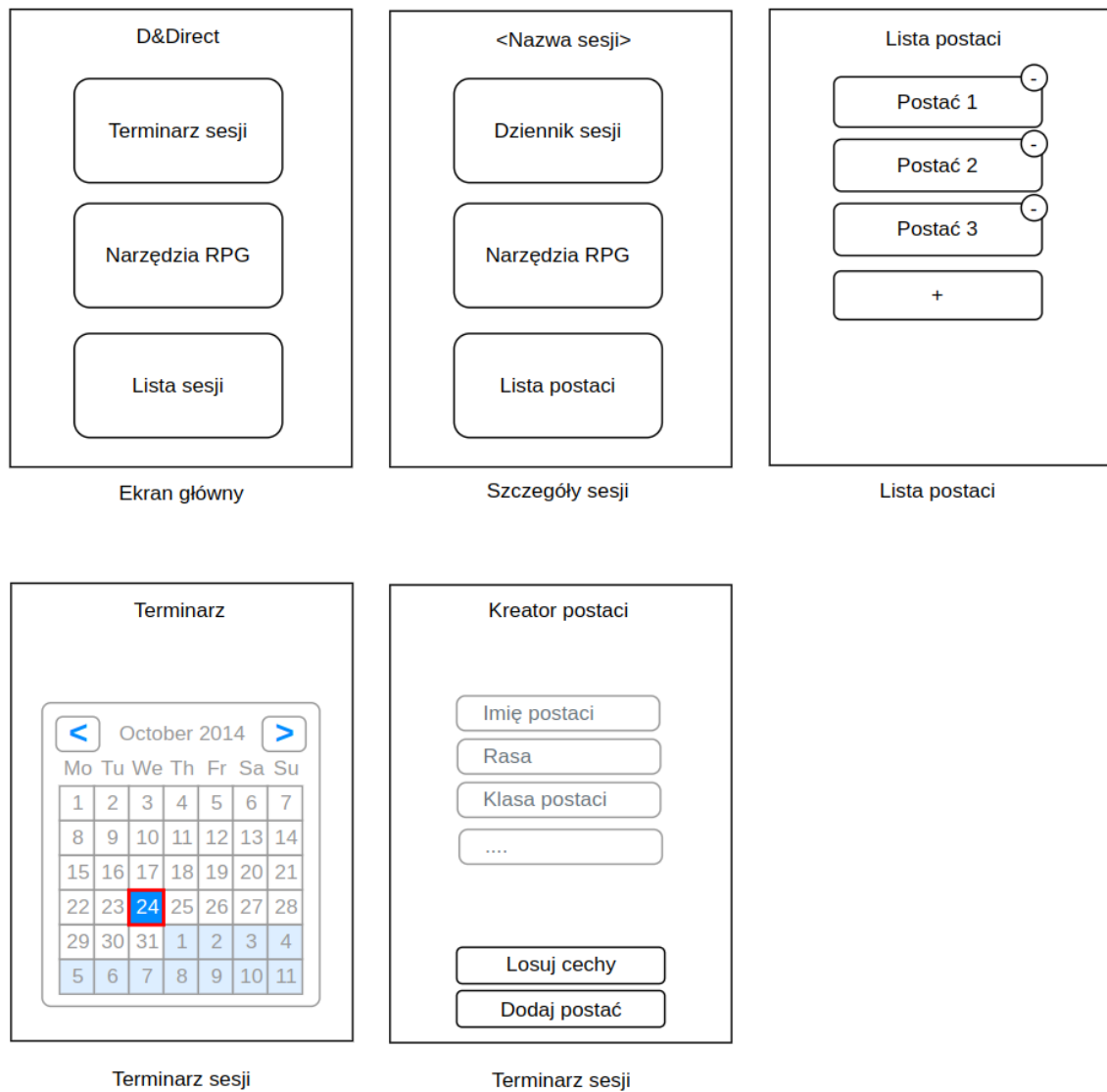


Przykład 2. Damian dodaje termin sesji



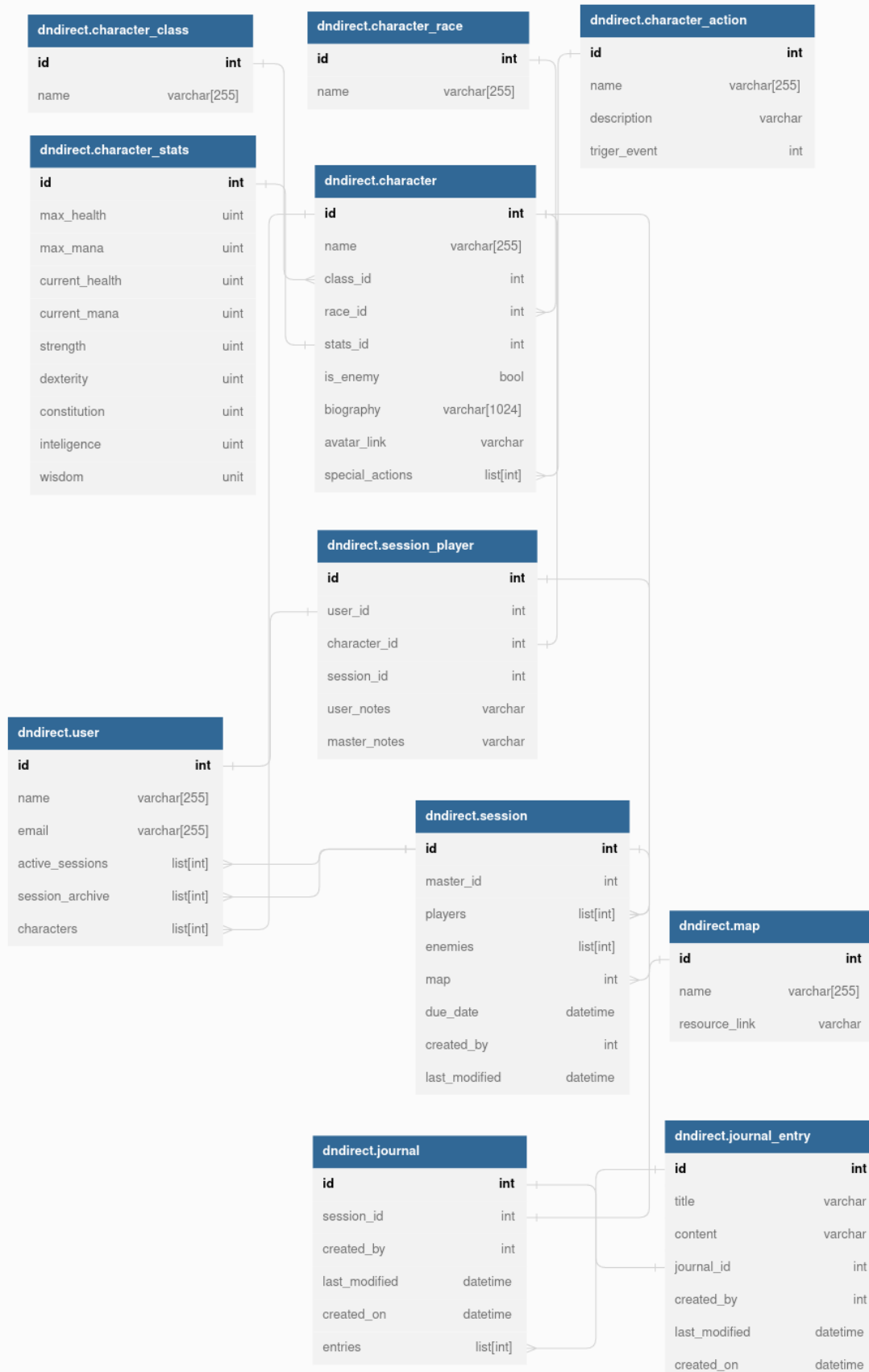
Przykład 3. Michalina korzysta z narzędzia do rzutu kostkami

2.2. Przykłady widoków na podstawie scenariuszy użycia



Przykład 4. Prototypy interfejsu użytkownika

3. Model konceptualny rzeczywistości



4. Projekt architektury systemu

System w podstawowej wersji będzie samowystarczalną aplikacją kliencką. Do zaimplementowania skorzystamy z frameworku ReactJS i całość napiszemy w JavaScript. Bazę danych postawimy na serwisie Microsoft Azure, a do komunikacji z bazą skorzystamy z języka msSQL.

W trakcie produkcji będziemy korzystać z narzędzia kontroli wersji git oraz kod umieścimy w prywatnym repozytorium na portalu Github. Ponadto wykorzystamy Github Actions do automatycznego testowania kodu, ciągłej integracji oraz ciągłego wdrażania zmian.

5. Identyfikacja zagrożeń

Największym problemem może być kwestia przejrzystości interfejsu użytkownika. Musi on zostać zaprojektowany tak, by był przystępny dla jak największej ilości osób mimo mnogości dostępnych opcji.

6. Porównanie z tablicą koncepcyjną i specyfikacją wymagań

Po utworzeniu bardziej szczegółowej koncepcji systemu i przemyśleniu projektu architektury systemu uważamy, że całkowicie pokrywa się ona z przyjętą wcześniej specyfikacją wymagań i ideą zawartą w tablicy koncepcyjnej. Uznajemy ramy czasowe i finansowe za realne i wystarczające do zrealizowania projektu.