

Tytuł pracy magisterskiej

Stanisław Zawadzki

Wrocław, 5 kwietnia 2011

Spis treści

1	Wstęp	3
1.1	Słownik	3
2	Terminologia szachowa	4
2.1	Szachownica, figury i zasady ich posunięć	4
2.1.1	Pionki	5
2.1.2	Lekkie figury	6
2.1.3	Ciężkie figury	7
2.1.4	Król	7
2.2	Posunięcia niezwykłe	7
2.2.1	Bicie w przelocie	8
2.2.2	Roszada	8
2.2.3	Promocja	8
2.3	Początek, przebieg i koniec rozgrywki	9
2.3.1	Rozpoczęcie rozgrywki	9
2.3.2	Różne sposoby zakończenia partii	9
2.4	Dodatkowe informacje o partiach szachowych	10
2.4.1	Czas rozgrywki	10
2.4.2	Zapis rozgrywki	10
2.5	PGN i FEN - sposoby zapamiętywania partii	11
2.6	Oznaczenia figur szachowych	11
2.7	Szachownice elektroniczne	12
2.8	Silniki szachowe	13
3	Ewaluacja dotychczasowych rozwiązań	16
3.1	Ankiety na temat istniejących rozwiązań	16
3.2	Kafejki szachowe	16
3.3	Programy szachowe	17
3.4	Chessbomb - wady i zalety serwisu	18
3.5	Cechy wspólne dotychczasowych rozwiązań oraz ich ocena	20

4	Projektowanie własnego interfejsu	22
4.1	Rodzaje użytkowników	22
4.2	Przypadki i scenariusze użycia oraz papierowy interfejs	23
4.2.1	Scenariusze użycia	23
4.2.2	Przypadki użycia	24
4.2.3	Papierowy prototyp	24
4.2.4	Wnioski z testów na papierowym prototypie oraz ich wpływ na kształt interfejsu	25
4.3	Wygląd interfejsu	25
4.4	Dobór kolorów interfejsu	26
4.5	Rodzaje figur i szachownic	26
5	Implementacja programu	26
5.1	Technologia	26
5.1.1	Ruby on Rails	26
5.1.2	JRuby on Rails	26
5.1.3	Dodatkowe biblioteki	27
5.2	Struktura programu	27
5.2.1	Funkcjonalności	27
5.2.2	Struktura bazy danych	28
5.3	Fragmenty najciekawsze algorytmicznie	29
6	Ewaluacja programu	29
6.1	Wyniki ewaluacji programu	29
6.2	Wnioski i ulepszenia wynikające z ewaluacji	29
6.3	Raport z produktu końcowego	29
7	Podsumowanie	29

1 Wstęp

1.1 Słownik

W trakcie tej pracy będzie przewijać się sporo terminów dotyczących szachów i programów szachowych. W tym punkcie wyjaśnię znaczenie owych słów, a także podam listę wyrażen do nich synonimicznych.

- **Szachy** - gra planszowa rozgrywana na szachownicach przy użyciu figur szachowych;
Synonim: królewska gra;
- **Szachownica** - plansza, na której rozgrywa się partię szachów. Podzielona na 64 pola - 32 białe i 32 czarne.
- **Figura szachowa** - Jedna z sześciu możliwych typów figur (pionek, goniec, skoczek, wieża, hetman, król). Zawodnicy przemieszczają je podczas swoich posunięć
Synonim: bierka szachowa
- **Posunięcie** - Zmiana położenia figury szachowej wykonana przy zachowaniu zasad poruszania się tych figur
Synonim: ruch, półruch
- **Szach** - Zaatakowanie króla przeciwnika, zmuszające go do natychmiastowej reakcji
Synonim: atak na króla
- **Mat** - Szach, po którym nie można nie utracić swojego króla. Kończy natychmiast partię jako wygrana zawodnika, który wykonał posunięcie matujące
- **Wygrana** - Aby wygrać partię należy zamatować przeciwnika lub skłonić go do dobrowolnego poddania partii. Zawodnik wygrywający otrzymuje jeden punkt, jego przeciwnik nie zyskuje żadnych punktów.
- **Remis** - Uznanie partii za nierozstrzygniętą. Skutkuje podziałem punktów między obu zawodników (oba otrzymują wówczas $\frac{1}{2}$ punktu).
Synonim: podział punktu
- **Propozycja remisu** - Dobrowolna oferta składana w dowolnym momencie gry przez jednego przeciwnika drugiemu. Jej zaakceptowanie oznacza podział punktu, jej odrzucenie nie wpływa na przebieg rozgrywki
- **Reklamowanie remisu** - Istnieją sytuacje, w których można wnioskować o odwrotne przyznanie remisu. Jeśli pozycja spełnia opisane w przepisach kryteria, wówczas sędzia może orzec podział punktu. W przeciwnym razie partii toczy się dalej.
- **Czas rozgrywki** - Choć nie jest to wymagane przepisami, w praktyce partie mają swój limit czasu. Partia trwa wówczas 2X, gdzie X oznacza czas dla jednego zawodnika. Czas zawodnikowi płynie wówczas gdy jest kolej na jego posunięcie. Do

odmierzania czasu korzysta się z zegarów szachowych. W momencie jak zawodnikowi kończy się czas do namysłu przegrywa on partię.

Synonim: tempo partii

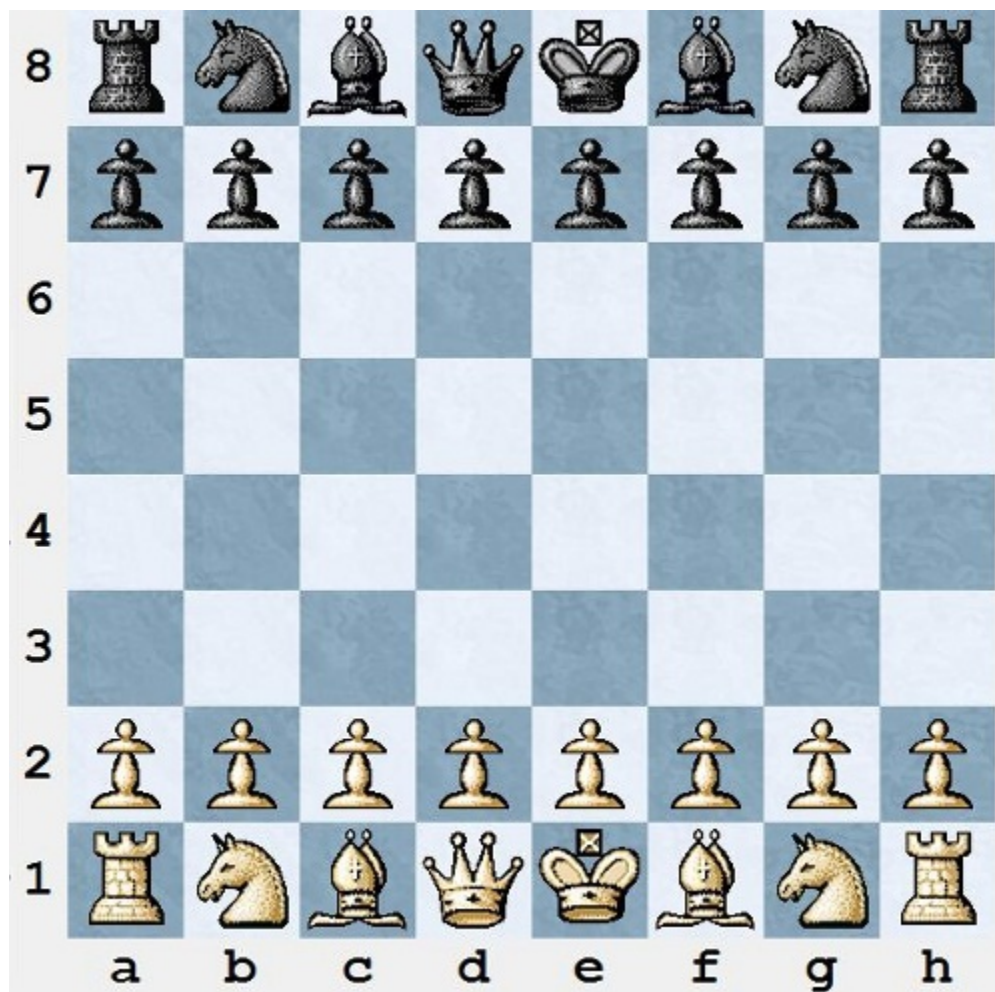
- **Zegar szachowy** - Urządzenie składające się z dwóch czasomierzy i przełącznika między nimi. Początkowo czas na obu licznikach jest ten sam, jednak upływa on różnie u obu zawodników - w zależności od czasu spędzonego na namyśle nad swoimi posunięciami.
- **Czas dodawany** - W niektórych przypadkach wybierane jest tempo gry, które modyfikuje wartość czasu na zegarze po wykonaniu posunięcia. Jest to zazwyczaj wartość pomiędzy trzema a trzydziestoma sekundami. Dodaje się ona automatycznie po przełączeniu dźwigni zegara.
- **Zawodnik grający białymi** - Gracz rozpoczynający grę.
Synonimy: Pierwszy zawodnik, gracz białymi, białe
- **Zawodnik grający czarnymi** - Gracz wykonujący posunięcia jako drugi.
Synonimy: Drugi zawodnik, gracz czarnymi, czarne
- **Silnik szachowy** - Program oceniający pozycję analizujący możliwe przebiegi partii.
Synonimy: Program szachowy, silnik, silnik analityczny
- **Zapis partii** - Zapisanie ciągu wszystkich wykonanych posunięć wraz z opisem zawodników i zawodów
- **Szachownica elektroniczna** - Specjalna szachownica pozwalająca na automatyczne zczytywanie przebiegu partii

2 Terminologia szachowa

Aby móc rozmawiać o szachowych interfejsach trzeba posiadać podstawową wiedzę na temat zasad królewskiej gry. W tym rozdziale postaram się opisać każdą możliwą sytuację w partii szachowej, ale także opiszę specyfikę sprzętu i oprogramowania, które będą kluczowe dla mojej aplikacji. Rozpocznę od opisu ruchu figur, później omówię posunięcia nadzyczajne a na końcu opiszę wszystkie inne aspekty, które są kluczowe w przypadku implementowania interfejsu szachowego. Na końcu zostaną opisane szachownice elektroniczne oraz silniki szachowe, które to umożliwią mojemu programowi jego funkcjonowanie.

2.1 Szachownica, figury i zasady ich posunięć

Choć zasady gry w szachy są wielu osobom znane, na potrzeby tej pracy chciałbym przytoczyć wszystkie informacje, które mają znaczenie dla interfejsów szachowych. Plansza, na której odbywa się rozgrywka jest nazywana szachownicą. Jest to obszar podzielony na 64



Rysunek 1: Pozycja początkowa

pola (8x8) numerowane w wierszach liczbami (1-8), w kolumnach zaś literami (a-h). W terminologii szachowej zarówno na wiersze, jak i na kolumny mówi się "linie". Patrząc z perspektywy gracza grającego białymi w lewym dolnym rogu jest pole a1, patrząc z perspektywy jego przeciwnika jest to pole h8. Pola są dwóch rodzajów - białe i czarne. Pola jednego koloru sąsiadują ze sobą po skosie i **zawsze pole a1 jest czarne**. Fizycznie nie ma wymagania, by używać dokładnie tych kolorów, więcej, w rzeczywistości czarne pola są najczęściej ciemnobrązowe, białe zaś mają kolor jasnożółty.

Jak mamy już ustawioną pozycję początkową musimy teraz rozpatrzyć zasady poruszania danych figur

2.1.1 Pionki

W przypadku pionków sytuacja jest dość prosta. Te najliczniej reprezentowane figury mogą się przesuwać wyłącznie do przodu. Kiedy przypada kolej danego zawodnika może on

wykonać pionkiem jedno z trzech możliwych posunięć:

1. Przesunąć pionka o jedno pole do przodu (t.j. wzdłuż kolumny)

Jest to podstawowe posunięcie możliwe do wykonania pionkiem. Można je wykonać wtedy, gdy pole na które zmierza pionek nie jest zajęte przez inną figurę (zarówno swoją, jak i cudzą).

2. Przesunąć pionka o dwa pola do przodu

Posunięcie to można wykonać tylko wtedy, gdy pionek zajmuje początkową pozycję. Jak w poprzednim przykładzie pole, na które zmierza pionek, jak i pole pośrednie nie może być zajęte przez żadną figurę.

3. Zbić figurę przeciwnika poruszając się po skosie do przodu

Pionki jako jedyne szachowe figury biją w inny sposób niż się poruszają. W zwyczajnej sytuacji pionki poruszają się wzdłuż kolumn. Jedynym sposobem, aby pionek przeszedł na sąsiednią linię jest zabicie innej figury. Pionki biją "po skosie", czyli jeśli pionek białych stoi na polu e3, to może zbić figurę, która stoi na d4 lub f4. W przypadku gdyby na e3 stał pionek czarnych, mógłby on zbić figurę na polu d2 lub f2. W wyniku zbiccia stawiamy pionka na polu dotychczas zajmowanym przez daną figurę, tamtą zaś usuwamy z szachownicy

Dodatkowym i bardzo rzadkim ruchem jest tzw. „bicie w przelocie”, o którym napiszę w dziale "Posunięcia niezwykłe"

2.1.2 Lekkie figury

W nomenklaturze szachowej mianem „figury lekkiej” nazywamy gońce i skoczki. W tym punkcie zajmiemy się zdefiniowaniem sposobu poruszania się przez te dwie figury. Goniec, zwany czasem błędnie lafrem, porusza się zawsze po skosie. Wynika z tego, że jeśli początkowo figura ta jest ustawiona na polu białego koloru, to do samego końca rozgrywki będzie się poruszała wyłącznie po polach tego właśnie typu. Goniec nie może „przeskoczyć nad przeszkodą”. Stąd wynika, że może poruszyć się maksymalnie tak daleko, jak dużo ma wolnego miejsca po diagonalu¹. Goniec może zbić pierwszą figurę, która staje na jego drodze.

Skoczek jest jedyną figurą, która może przeskoczyć nad innymi figurami. Ruchy skoczka jest najłatwiej porównać do litery „L”. Może on wykonać posunięcie w każdą stronę, pod warunkiem iż będzie to ruch opisany jako "dwa pola do przodu i jedno w bok". W przypadku skoczka zwrot „do przodu” oznacza dowolny z czterech kierunków. Bicie skoczkiem polega na tym, że skoczek może zabrać figurę z tego pola, na które uda mu się wskoczyć.

¹W nomenklaturze szachowej każda ukośna linia mająca przynajmniej trzy pola jest diagonalą

2.1.3 Ciężkie figury

Podobnie jak w przypadku lekkich figur termin „ciężkie figury” określa dwa rodzaje szachowych figur: wieżę i hetmana. Figury ciężkie w zdecydowanej większości przypadków są silniejsze od pozostałych figur.

Posunięcia wieży są bardzo kanoniczne - porusza się ona wzdłuż pionowych i poziomych linii. Podobnie jak w przypadku gońca może ona bić pierwszą figurę, która wystąpi na jej drodze. Innym ruchem, który może ona wykonać, jest roszada opisana w dziale posunięć niezwykłych.

Jak porusza się hetman chyba nie muszę opisywać z jednego bardzo prostego powodu. Problem ośmiu hetmanów jest na tyle znanym zagadnieniem algorytmicznych, że każdy absolwent studiów informatycznych musiał się z nim zapoznać. Skrótem opisując - hetman jest hybrydą wieży i gońca - łączy wszystkie cechy obu tych figur (poza możliwością roszady)

2.1.4 Król

Król jest, w zależności od punktu widzenia, albo najsłabszą, albo najsilniejszą figurą. Porusza się on w bardzo prosty sposób - może przejść na każde z okolicznych pól. Jedynym dodatkowym posunięciem w jego arsenale jest wzmiankowana wcześniej i opisana w następnym podrozdziale roszada. Jest jednak bardzo ważna cecha, która odróżnia króla od każdej innej figury. Monarcha nie może zostać umieszczony na polu atakowanym przez chociaż jedną figurę rywala. Tak samo trzeba zawsze coś przedsięwziąć, jeśli przeciwnik wykonując ruch zaatakował naszego króla. W tym przypadku trzeba wykonać jeden z trzech manewrów

- odejść królem na nieatakowane pole
- zbić atakującą figurę
- przesłonić obszar działania atakującej figury

Król jest jedyną figurą z naszego arsenału, której nie możemy stracić. W momencie gdy monarcha zostaje tak osaczony, że nie może uniknąć zagłady, następuje koniec partii. Ale o tym będzie w dalszych podrozdziałach.

2.2 Posunięcia niezwykłe

W poprzednim punkcie przedstawiłem wszystkie zwyczajne sposoby na wykonanie posunięcia. W tym punkcie opisuję przypadki rzadkich posunięć niezwykłych, które są odstępstwem od podanych wcześniej standardów. Niektóre z nich zdarzają się bardzo rzadko, niektóre z nich w prawie każdej szachowej partii.

2.2.1 Bicie w przelocie

Bicie w przelocie (ang. *en passant*) jest najrzadszym szachowym posunięciem. Występuje ono tylko w jednym momencie. Warunkiem koniecznym i wystarczającym, aby można było zagrać ten ruch jest fakt, iż pionek przeciwnika ruszył się o dwa pola do przodu, zaś nasz pionek atakuje pośrednie pole między początkowym o końcowym ustawinieniem pionka rywala. Aby dokonać zbitia przesuwamy pionka na owo pole pośrednie i zdejmujemy pionka rywala (jest to jedyny przypadek w grze, że bije się figurę z pola, na które się nie postawiło swojej figury). Ważną sprawą jest to, że ten ruch można wykonać tylko bezpośrednio po ruchu pionem o dwa pola. Później takie bicie jest niezgodne z przepisami.

2.2.2 Roszada

W przeciwieństwie do bicia w przelocie roszada, choć jest posunięciem niezwykłym, zdarza się bardzo często. Jest ona jedynym szachowym posunięciem, które wprawie w ruch więcej niż jedną figurę. Polega ona na jednoczesnym przesunięciu króla i wieży po pierwsze linii (odpowiednio ostatniej). Muszą być jednak spełnione następujące warunki

1. Wszystkie pola między królem a wieżą muszą być puste,
2. Zarówno król, jak i wieża nie mogły wcześniej wykonać ani jednego posunięcia,
3. Król nie jest atakowany przed wykonaniem roszady, nie będzie zaatakowany po wykonaniu roszady i nie przekroczy zaatakowanego pola w trakcie jej wykonywania.

Oczywiście drugi z tych punktów determinuje, że roszadę można wykonać maksymalnie raz w czasie całej partii.

Ponieważ są dwie wieże na szachownicy to rozróżniamy dwa typy roszad.

- roszada krótka (zdarza się częściej) - król przechodzi z pola e1 na pole g1, wieża przesuwa się z h1 na f1
- roszada długa - król przechodzi z pola e1 na pole c1, wieża przesuwa się z pola a1 na pole d1 (zauważmy, że w przypadku tej roszady pole b1 **może** być atakowane).

2.2.3 Promocja

Promocji dokonuje się z pomocą przesunięcia pionka aż do końcowej linii szachownicy (ósmej w przypadku białego pionka, pierwszej w przypadku czarnego). W momencie gdy pionek osiąga skraj szachownicy **musi** on być zmieniony na hetmana, albo wieżę, albo gońca, albo skoczka **tego samego koloru!** Dozwolone jest posiadanie więcej figur, niż określa to stan początkowy (czyli można mieć na szachownicy nawet dziewięć hetmanów). Promocji można dokonać każdym pionkiem i cała czynność, czyli postawienie pionka na ostatniej linii oraz zamienienie go na figurę trwa dokładnie jeden ruch.

2.3 Początek, przebieg i koniec rozgrywki

W poprzednim punkcie opiszę warunki początkowe i końcowe rozgrywki szachowej

2.3.1 Rozpoczęcie rozgrywki

Na początku figury są rozstawione w sposób przedstawiony na rysunku nr XXX. Po zwyczajowym przywitaniu się z przeciwnikiem następuje uruchomienie zegara szachowego i białe wykonują swój pierwszy ruch. Od tej chwili obaj zawodnicy wykonują na zmianę po jednym posunięciu, korzystając z opisanych wcześniej zasad. W momencie gdy zawodnik zaatakuję króla rywala, powinien powiedzieć "szach", aby ów wiedział, że ma obowiązek zaradzeniu temu problemowi. W zasadach, to w rozgrywce na żywo jest obowiązek powrotu do ostatniej poprawnej pozycji (pozycja jest poprawna, jeśli wszystkie dotychczasowe ruchy są zgodne z zasadami szachowymi). W szachach wirtualnych komputer powinien nie dopuścić do wykonania niedozwolonego posunięcia. Posunięcia są wykonywane na zmianę aż do momentu, aż w jakiś sposób nadejdzie koniec partii. A dróg, aby zakończyć partię jest naprawdę wiele

2.3.2 Różne sposoby zakończenia partii

Partia może zakończyć się na dwa sposoby - remisem lub zwycięstwem jednej ze stron. Mniej opcji jest w przypadku porażki. Zakończnie partii określają przepisy Międzynarodowej Federacji Szachowej (FIDE):

ARTYKUŁ 5. PARTIA ZAKOŃCZONA

5.1. (a) Partię wygrywa zawodnik, który zamatował króla przeciwnika. Mat natychmiast kończy partię pod warunkiem, że pozycja matowa powstała w wyniku prawidłowego posunięcia.

(b) Partię wygrywa zawodnik, którego przeciwnik oświadczył, że poddaje się. Poddanie natychmiast kończy partię.²

5.2 (a) Partia kończy się remisem, gdy zawodnik będący na posunięciu nie może wykonać żadnego prawidłowego ruchu, a jego król nie jest szachowany. Taką pozycję określa się słowem „pat”. Pat natychmiast kończy partię, pod warunkiem, że pozycja patowa powstała w wyniku prawidłowego posunięcia.

(b) Partia kończy się remisem, jeśli powstaje pozycja, w której żaden z zawodników nie może zamatować króla za pomocą jakiegokolwiek serii prawidłowych posunięć. Pojawienie się tzw. „martwej pozycji” też natychmiast kończy partię, pod warunkiem, że powstała ona w wyniku prawidłowego posunięcia. (Patrz art. 9.6)

(c) Partia kończy się remisem w wyniku uzgodnienia pomiędzy obydwoma zawodnikami w trakcie partii. Zgoda na remis natychmiast kończy partię. (Patrz art. 9.1).

(d) Partia może zakończyć się remisem, jeżeli identyczna pozycja ma pojawić się na szachownicy przynajmniej po raz trzeci lub już pojawiła się przynajmniej trzykrotnie. (Patrz

²Przepisy gry od 01.07.2009 wg Światowej Federacji Szachów

art. 9.2).

(e) Partia może zakończyć się remisem, jeżeli obaj zawodnicy przynajmniej w ostatnich 50 posunięciach nie wykonali żadnego ruchu pionkiem, ani nie pobili żadnej bierki. (Patrz art. 9.3).

Trzeba zawsze pamiętać o każdym z wymienionych sposobów i posiadać narzędzia pozwalające sprawdzić, czy strony mogą zgodzić się na podział punktu. W ostatnich latach spora część zawodów szachowych dodała zasadę zakazu remisu w mniej niż 30 posunięć, jednak wciąż jest to jedynie wewnętrzny przepis - oficjalnie wystarczy, że zawodnicy zagrają dwa półruchy³ Nie licząc sytuacji zupełnie losowych⁴ partie mogą kończyć się wyłącznie w sposoby określone w wyżej przytoczonych przepisach.

2.4 Dodatkowe informacje o partiach szachowych

Pisząc o partiach szachowych należy przytoczyć jeszcze kilka terminów, które dotyczą każdej rozgrywki.

2.4.1 Czas rozgrywki

W zdecydowanej większości przypadków partie szachowe są grane z limitem czasu. Może to być zaledwie minuta dla zawodnika na partię, a może to być kilka godzin. W ostatnich latach coraz bardziej popularna jest opcja dodawania czasu po wykonaniu każdego posunięcia. Zawodnicy mogą zużyć swój czas w zależności od potrzeb na analizowanie swoich posunięć. W momencie gdy wykonają ruch przełączają upływ czasu na konto rywala. W przypadku, jeśli zabraknie czasu do namysłu, partia jest uznana za przegraną.

2.4.2 Zapis rozgrywki

W partiach granych dłuższym tempem (minimum godzina dla zawodnika) jest obowiązek prowadzenia zapisu partii. Zapis służy zapamiętywaniu stanu partii po każdym wykonanym posunięciu, aby w każdej chwili można było prześledzić historię rozgrywki. Ogólnoświatową konwencją jest stosowanie tzw. skróconej notacji algebraicznej. Na początku zapisywana jest pierwsza litera figury⁵, później zaś pole, na które ona zmierza. Jeśli jest to nierozstrzygające dopisuje się jeszcze informację, która z figur⁶ W świecie rzeczywistym odbywa się to z użyciem papierowych blankietów, w świecie programów komputerowych przechowuje się wiadomości o partii z użyciem plików o określonej strukturze.

³Wcześniej wg definicji partia nie jest jeszcze uznana za rozpoczętą

⁴Zdarzały się nawet przypadki śmierci przy szachownicy

⁵W niektórych językach, jak n.p. w angielskim jest to nie zawsze możliwe. Wówczas stosuje się inne oznaczenie

⁶Na przykład Sde2 - skoczek z linii 'd' idzie na pole e2

2.5 PGN i FEN - sposoby zapamiętywania partii

Od kiedy tylko informatyka włączyła się komercyjnie w świat szachów powstało pytanie jak skutecznie zapisywać i odczytywać pojedyncze rozgrywki szachowe. W tym celu powstał z czasem format **PGN (Portable Game Notation)**⁷, w którym w prosty sposób można zapisać wszystkie informacje o partii szachowej.

Jest to format tekstowy, czytelny dla bardziej doświadczonego szachisty. Nawet jeśli nie dysponujemy żadnym programem, który graficznie przedstawiłby daną rozgrywkę, bez najmniejszych problemów można otworzyć ten plik w edytorze tekstowym i przeczytać nagłówek i posunięcia. Format ten sprawdza się dobrze w przypadku pojedynczych partii, jednak w przypadku wielkich baz (kilka milionów partii) staje się bardzo nieefektywny z uwagi na zerową kompresję.⁸

FEN (Forsyth-Edwards Notation) służy nie do zapisu partii, ale do zapamiętania konkretnej sytuacji na szachownicy. Czasem bowiem mamy do czynienia z sytuacją, kiedy nie potrzebujemy wcześniejszego przebiegu rozgrywki i interesuje nas tylko konkretna w niej pozycja. Początkowo wydaje się to trywialnym zadaniem, jednak po głębszym namyśle odkrywamy problemy - jak zapisać, czy któraś ze stron może jeszcze zrobić roszadę (nie wynika to z samego ustawienia, bo król mógł używając przynajmniej dwóch posunięć raz jeszcze znaleźć się na pozycji początkowej), czy też możliwość bicia w przelocie? Do tego trzeba zadbać o zwięzłość tego zapisu. Notacja FEN opisuje linia po linii, każde pole na szachownicy, zliczając pola puste.

Przykładowo - pozycja z pięcioma figurami, białym królem na polu g5, hetmanem na polu e6 oraz czarnymi figurami na polach g7(król), f7 (wieża) i h7 (pionek) wygląda następująco "8/5rkp/4Q3/6K1/8/8/8/8 w - - 0 1" Znak „/" oznacza nową linię, wielkie litery opisują białe figury, małe litery opisują figury czarne. Litera 'w' oznacza, że grę rozpoczynają białe. Myślniki świadczą o braku możliwości zrobienia roszady. W ten sposób, korzystając z zaledwie kilkunastu znaków opisaliśmy całą pozycję.

Z notacji FEN często korzysta się w plikach PGN, gdyż dzięki niej można rozpocząć zapis partii od dowolnej pozycji, a nie tylko początkowej.

2.6 Oznaczenia figur szachowych

W formacie PGN należy rozstrzygnąć w jaki sposób oznaczamy figury W szachach nie ma jednolitego międzynarodowego nazewnictwa, każdy kraj stosuje nazwy w swoim języku. Jedynym standardem jest omijanie przedrostka, w przypadku gdy dochodzi do posunięcia pionkiem. Oczywiście każdy profesjonalny szachista potrafi korzystać z nazw figur w języku angielskim⁹(król-K,hetman-Q,wieża-R,goniec-B, skoczek-N), jednak nie należy zakładać, że każdy użytkownik będzie posiadał tę wiedzę. Rozwiązaniem jest korzystanie z

⁷można ją znaleźć m.in. pod adresem <http://pgn.freesevers.com/Standard.txt>

⁸Wówczas stosowane są inne, komercyjne formaty, jednak nie mają one żadnego związku z tematem mojej pracy

⁹Nie należy jednak tego mylić z tzw. notacją angielską, która jest obecnie już nieużywanym sposobem zapisywania posunięć

piktogramów, które nie tylko już są pewnym standardem¹⁰, ale są także graficznie zbliżone do rzeczywistych figur.

2.7 Szachownice elektroniczne

Jeśli myślimy o transmisjach partii szachowych na żywo musimy także zrozumieć sposób, w jaki posunięcia szachowe są transmitowane w Internecie. W tym celu korzysta się ze specjalnego sprzętu zwanego szachownicami elektronicznymi¹¹. Z wyglądu przypominają one zwykłe szachownice, jednak podłączone są one przewodowo¹² do komputera, który przy pomocy programu autorstwa firmy DGT odczytuje sygnały z szachownic i interpretuje je jako posunięcia. Wówczas można już te partie zapisywać do plików i wysyłać na serwery do odczytu przez programy prowadzące transmisje.

Podczas prowadzenia transmisji z użyciem szachownic elektronicznych zdarzają się błędy, które można podzielić na kilka kategorii:

- Błędy fizyczne
Na salach turniejowych (wyłączywszy nieliczne superturnieje) nie ma możliwości, aby kable poprowadzić pod podłogą lub w inny sposób uniemożliwiający przypadkowe ich zerwanie. Z tego powodu trzeba brać pod uwagę, iż któryś z zawodników¹³ spowoduje zerwanie kabla z transmisji. Niesprawne lub niepoprawnie przekazujące dane zegary także są częstą przyczyną błędów w przypadku oglądania partii on-line
- Błędy szachistów
Doświadczeni szachiści bardzo rzadko popełniają błąd polegającym na wykonaniu nieprawidłowego posunięcia¹⁴. Mimo to, zwłaszcza w przypadku ograniczonego czasu do namysłu, zdarza się sytuacja, w której zawodnik wykona nielegalne posunięcie¹⁵. Spora część programów transmitujących partie nie bierze pod uwagę takiej sytuacji. Powoduje to, że dalsza część partii jest odrzucana jako nieprawidłowa i nie jest pokazywana przez aplikację.
- Błędy sędziego/technika zawodów
Kolejną osobą, która może popełnić błąd jest sędzia lub technik zawodów. Mając dostęp do panelu administrowania szachownicami może wpisać źle dane partii, może także wpisać zły wynik. Może oczywiście również źle skonfigurować sieć, przez co pliki nie będą prawidłowo umieszczane na serwerze.
- Błąd „późniejszej analizy”
Jest to bardzo częsty błąd, który musi być starannie usuwany przez sędziego. Bie-

¹⁰Istnieją choćby w domyślnych czcionkach systemów biurowych jak Microsoft Word czy OpenOffice

¹¹Aktualnie rynek szachownic elektronicznych zmonopolizowany jest przez holenderską firmę DGT

¹²Przez gniazdo COM

¹³W czasie namysłu rywala wielu szachistów wstaje od swojej partii i spaceruje po sali gry

¹⁴Amatorzy rzadko grają na szachownicach elektronicznych, gdyż jest to drogi sprzęt

¹⁵Najczęściej jest to wykonanie nieprawidłowej roszady, choć zdarzają się też inne możliwości jak przypadkowe złapanie złej figury

rze się on z zachowania szachistów, którzy po zakończonej partii często przesuwają figury, aby sprawdzić "co by było, gdyby?". Komputer nie otrzymał informacji, że obaj zawodnicy już słownie uzgodnili wynik partii. Co prawda teoretycznie unika się tego poprzez specyficzne ustawienie królów zaraz po zakończeniu partii¹⁶, ale w ferworze walki szachiści bardzo często o tym zapominają. Czasem wynika problem, jeśli w końcowej pozycji postawienie króla na jednym z wyróżnionych pól jest możliwe. Analizy po zakończeniu partii są bardzo często algorytmicznie nierozróżnialne od samej partii, zatem wówczas sędzia musi ręcznie, na podstawie papierowych zapisów partii zweryfikować jej zakończenie.

Szachownice elektroniczne są konieczne do transmisji na żywo i choć w dalszej części pracy przyjmuję, że działają sprawnie, to podczas tworzenia programu należy zapewnić podstawową obsługę tych błędów.

2.8 Silniki szachowe

Silniki szachowe są programami, które jako dane wejściowe otrzymują pozycję, zaś jako rezultat zwracają ocenę tej pozycji wraz z propozycjami najlepszych posunięć. Nie należy jednak podchodzić do nich w sposób jaki podchodzi się do zwyczajnych programów - ich wynik w zasadzie nigdy nie jest rozstrzygający i pewny, a ich działanie może być pod względem praktycznym nieskończone. Bierze się to z olbrzymiej ilości możliwych kontynuacji z danej pozycji. Mimo iż szachownica posiada zaledwie 64 pola, a każda z figur posiada ograniczone pole działania przestrzeń możliwych sekwencji ruchów jest olbrzymia.

Szachy są grą skończoną. Wynika to jednoznacznie z trzech obserwacji:

1. Ilość figur na szachownicy może się jedynie zwiększać¹⁷
2. Każdy pionek ma skończoną możliwą liczbę posunięć
3. W przypadku zagrania 50 posunięć bez bicia lub posunięcia pionkiem ogłaszany jest remis.

Z punktów 1. i 2. wynika, iż ilość bić i posunięć pionkiem jest ograniczona liczbą $30(\text{zbić}) + (2 \cdot 8 \cdot 6)$ (możliwe posunięcia wszystkich pionków), czyli łącznie możemy mieć maksymalnie 126 podobnego typu posunięć, co w połączeniu z punktem trzecim daje nam maksymalnie $50 \cdot 126 = 6300$ posunięć w jednej partii szachowej, co udowadnia, że szachy są grą skończoną.

Mimo to, że każda partia szachowa musi się kiedyś zakończyć, przestrzeń możliwych kontynuacji jest olbrzymia. Szacuje się, że możliwych sekwencji jest około 2^{100} , co jest

¹⁶Biały król na e4, czarny na d5 w przypadku wygranej białych, biały król na e4, czarny król na e5 w przypadku remisu oraz biały król na d4, czarny na e5 w przypadku wygranej czarnych

¹⁷Promocja może zwiększyć jakość figur, jednak nie zwiększy ich ilości

oczywiście poza zasięgiem zdolności obliczeniowej komputerów. Nawet jeśli podejź się do sprawy bardziej praktycznie, eliminując za pomocą heurystyk jawnie błędne kontynuacje, to ilość możliwości jest wyzwaniem nawet dla współczesnych procesorów. Mimo iż komputery w chwili obecnej bez problemu pokonują najlepszych szachistów¹⁸, to nie należy zwięć się iluzji o ich wszechpotędze. Programy osiągają przewagę nad ludźmi dlatego, że nie popełniają błędów i cały czas grają na bardzo wysokim poziomie. Cały czas jednak istnieją wyzwania, z którymi poradził sobie ludzki umysł, a maszyna wciąż nie potrafi.

Historia silników szachowych sięga lat 50-tych ubiegłego wieku, jednak dopiero w latach 80-tych maszyny zaczęły prezentować poziom, który pozwalał im na nawiązanie walki z profesjonalnymi szachistami. Lata dziewięćdziesiąte były czasem walki o supremację między ludźmi i maszynami, zaś pierwsza dekada obecnego stulecia doprowadziła do sytuacji, że człowiek jest już bez szans w pojedynku z maszyną. Prawda jest jednak taka, że podstawowe zasady programowania silników szachowych pozostały podobne do tych, które rozwijali uczeni w kilkadziesiąt lat temu.

Istnieją dwa podstawowe typy filozofii tworzenia silników szachowych. Pierwsze działają korzystając z algorytmów *brute-force*, drugie zaś dokonują wstępnej selekcji możliwych posunięć. Co może dziwić, przez wiele lat programy z pierwszej z tych grup osiągały lepsze wyniki. Dlaczego się tak działo? Odpowiedź jest prosta - bardzo trudno dokonać poprawnej selekcji posunięć. Nawet jeśli w 99% przypadków program dokona słusznej eliminacji, wciąż w wielu momentach partii nie będzie on rozpatrywał najsilniejszego posunięcia¹⁹. W takim przypadku zysk z eliminacji, czyli możliwość spojrzenia dalej w przyszłość jest niwelowany przez ominięcie kluczowych wariantów. Drugi sposób działa w sposób przypominający ludzkie myślenie, jednak przez lata doprowadzało to do sytuacji, że programy korzystające z heurystyk selekcji posunięć były mniej konsekwentne niż programy pierwszego rodzaju, zaś wciąż zbyt słabo naśladowały umysł profesjonalnego szachisty. Jednak koniec końców programistom udało się doprowadzić do sukcesu drugiej grupy programów.

Silniki szachowe są programami pisanymi bardzo niskopoziomowo - szybkie działanie jest kluczowe dla nich i każda zbędna instrukcja byłaby wykonywana miliony razy na sekundę, co w oczywisty sposób odbiłoby się na wynikach obliczeń. Działają one korzystając z algorytmów min-max, zwracając jako wynik ocenę pozycji po najlepszej według nich sekwencji posunięć. Kluczowe dla nich są algorytmy oceny pozycji - muszą one jako argument wziąć ustawienie figur i zwrócić wynik w postaci liczby. Umiejętność dobrego przełożenia rozmaitych przewag i słabości pozycji na zwyczajną cyfrę często decyduje o tym, czy program działa dobrze, czy przeciętnie. W tym etapie programowania bardzo często biorą udział prawdziwi szachiści, którzy korzystając ze swojej szachowej wiedzy oceniają, czy ważniejsza jest przewaga materialna, czy inne czynniki - tymczasowe czy też statyczne. Jako że szachiści są tylko ludźmi, to nawet mając wiele doświadczenia w tym sporcie czasem popełniają w tym mniejsze lub większe błędy. Ostatecznie ten, który połączy najsprawniej napisany algorytm obliczania drzewa przeszukiwań i oceniania pozycji w liściach, stworzy

¹⁸Dekadę temu Garri Kasparow toczył ciężkie boje z superkomputerem Deep Blue, w chwili obecnej pokonałby go zapewne komercyjny program uruchomiony na zwyczajnym komputerze domowym

¹⁹Selekcja nie odbywa się tylko w korzeniu obliczeń, ale w większości jego potomnych węzłów również

najlepszy program grający.

3 Ewaluacja dotychczasowych rozwiązań

Programy służące do przeglądu partii szachowych istnieją już od wielu lat. Każdy profesjonalny szachista, ale także spora liczba amatorów korzysta z całej gamy obecnych na rynku produktów. W zasadzie każdy szachista, który wyszukuje w Internecie informacje o szachach korzysta także z programów realizujących inne funkcje.

3.1 Ankiety na temat istniejących rozwiązań

3.2 Kafejki szachowe

Od początków Internetu ludzie uznali, że ciekawie jest wykorzystywać to medium do rozgrywek na odległość. Już w latach 80-tych powstały pierwsze technologie i protokoły pozwalające na rozgrywki poprzez sieć. Od początku obecnego wieku jest to rozrywka coraz popularniejsza i skupiająca ludzi z całego świata. Oferują one swoim użytkownikom możliwość gry w szachy²⁰ zarówno z ludzkimi rywalami, jak i czasem przeciwko komputerom szachowym. Użytkownicy mają najczęściej do wyboru opcje wyszukiwania przeciwnika wg ustalonych przez siebie kryteriów. W zależności od preferencji są to czasem osoby, które grają jedną partię w godzinę, czasem są to osoby wybierające błyskawiczne tempo gry. Cechą wspólną jest jednak praktyczne podejście do gry - muszą mieć możliwość sprawnego rozgrywania partii szachowej, podczas grania której rzadko przełączają się między oknami tudzież odchodzą od komputera.

Badania dotyczące kafejek szachowych przeprowadziłem na podstawie najpopularniejszego w Polsce serwisu na platformie kurnik.pl. Mimo iż profesjonalni szachiści omijają ten serwis z daleka, tysiące amatorów spędzają na nim godziny swojego czasu. W ramach badań przeprowadziłem ankietę, w której wzięło udział kilkudziesięciu zawodników korzystających w tej konkretnie kafejki. Pierwszą rzeczą, którą można zaobserwować jest to, iż szachiści niezbyt przejmują się szatą graficzną interfejsu. Wnioskuje to z tego, że mimo iż większość postronnych uważasz serwis za bardzo nieestetyczny, u szachistów znajduje on wysokie uznanie. Jest to dość ważna obserwacja, gdyż można z niej wyciągnąć wnioski przy projektowaniu grafiki serwisu. Choć nie oznacza to, że sprawę wyglądu można całkowicie pominąć, jednak nie należy na nią przeznaczać wielkich środków. Ważną kwestią, na jaką ankietowani odpowiadali, był sposób oznaczenia ostatniego posunięcia - większość osób uznała, że sposób w jaki na kafejce kurnik jest zadowolający, jednak istnieje także spora liczba osób, które nie było zadowolone z prezentowanego rozwiązania. Jest to dość istotna kwestia dla mojej aplikacji, gdyż w transmisji na żywo także warto w jakiś sposób podświetlić, jakie było ostatnie zagrane posunięcie. Różne serwisy implementują to w rozmaity sposób. Na serwisie ICC korzystającego z programu BlitzIn oznaczone są czerwonymi obwódkami wokół dwóch pól²¹. Podobny system jest używany w serwisie Chessbomb, o którym będę pisał nieco później. On także używa obwódek wokół pól, jedynie zaznacza je kolorem niebieskim. Kurnik funkcjonuje w podobny sposób, jednak zamiast obwódek

²⁰A czasem także w gry podobne do szachów

²¹Startowym i końcowym polem figury która wykonała ostatnie posunięcie

zmienia kolor całego pola na nieco ciemniejszy. Zupełnie inną filozofie prezentuje serwis PlayChess, który ostatnie posunięcie zaznacza za pomocą kolorowej strzałki. Ważną kwestią dla wielu wydaje się zapis partii, jednak w przypadku kafejek nie jest on aż tak istotny jak mogłoby się wydawać. Użytkownicy po skończonej partii rzadko zastanawiają się, gdzie popełnili błąd, częściej zaś siadają od razu do następnej rozgrywki. Wśród ponad sześćdziesięciu uczestników ankiety tylko jedna osoba zwróciła uwagę na to, że w podstawowym wyglądzie okna nie ma zapisu. Pozostałe kafejki udostępniają zapis w podstawowym widoku, jednak najwyraźniej rozwiązanie przyjęte w kurniku spotyka się z aprobatą graczy. Jako główną zaletę kafejki kurnik przedstawiana często była duża liczba osób. Wynika z tego, że wskazanym jest, aby użytkownicy także mieli kontakt z innymi osobami korzystającymi ze strony. Choć w przypadku kafejek szachowych, gdzie główną ideą jest rozgrywanie partii z innymi użytkownikami jest to bardziej oczywiste, jednak kontakt z innymi szachistami wydaje się wskazany.

Wgląd do świata kafejek szachowych jest istotny, nawet jeśli one nie są całkowicie pararelne z aplikacjami do oglądania partii na żywo. Mają inne cele, inne priorytety i inne sposoby przedstawiania. Kafejka musi zapewnić użytkownikowi wygodę w znalezieniu dopasowanego przeciwnika, musi działać w czasie rzeczywistym i dostarczać bardziej zabawę niż edukację. W kafejce szachowej dużo ważniejsze jest przedstawienie czasu i ostatniego posunięcia, mniej ważny jest zaś dostęp do zapisu posunięć. Mimo to sposoby oznaczania szachowni i figur są wspólne dla obu zagadnień i dlatego wiedza z rozwiązań występujących w kafejkach szachowych można wykorzystać podczas tworzenia innych serwisów.

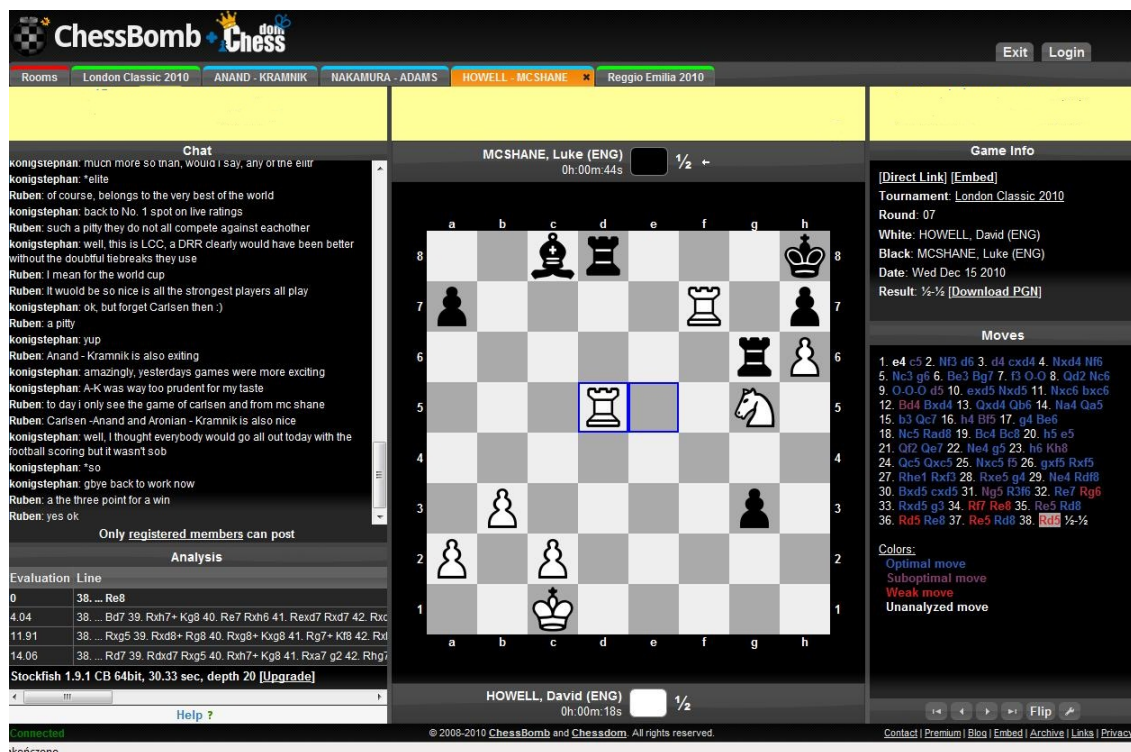
3.3 Programy szachowe

Profesjonaliści i półprofesjonaliści szachiści często korzystają z programów pomagających im zarówno w treningu, jak i wyszukiwaniu informacji o przeciwnikach. Jest to bardzo dobra grupa użytkowników do badań, gdyż najczęściej używają oni wybranych produktów w dłuższym okresie trwania swojej sportowej kariery. Ostatecznie przekłada się to na tysiące godzin spędzonych nad aplikacją. W przypadku tych programów sprawa się ma podobnie jak z kafejkami szachowymi - realizują one inne cele niż aplikacje służące relacji na żywo. Wspierają one mocno wyszukiwanie partii po kluczowych danych (czyli czynność zupełnie niespotykaną zarówno w kafejkach szachowych, jak i w programach do przeglądania partii) oraz działanie silników szachowych. Druga z tych funkcjonalności bardziej jest istotna dla mojej aplikacji.

Jak pisałem już w dziale dotyczącym silników, obliczenia przez nie prowadzone są bardzo kosztowne dla zasobów. Z tego powodu każdy program szachowy włącza silnik dopiero na wyraźne żądanie użytkownika. Doświadczony szachista najlepiej wie, w którym momencie partii należy włączyć program analizujący.²² Programy po włączeniu pokazują kilka najlepszych według nich opcji w danej pozycji²³ posortowanych względem oceny końcowej pozycji po wykonaniu ciągu najlepszych wg komputera posunięć. Można modyfikować dane

²²W naszej aplikacji nie będziemy mieli takiego luksusu - wszystko będzie się odbywało automatycznie

²³Czasem zdarza się tak, że kilka ruchów jest równorzędnie dobrych



Rysunek 3: Serwis ChessBomb - widok na szachownicę

pomocą silnika szachowego aktualną pozycję. Ta ostatnie funkcjonalność jest nowością - wcześniejsze przeglądarki on-line nie oferowały natychmiastowej analizy pozycji, co zmuszało użytkowników do korzystania ze swoich lokalnie zainstalowanych programów. Obliczenia, za pomocą których ocenia się pozycję szachową są bardzo wymagające pod względem obliczeniowym.²⁵ Użytkownik po swojej stronie może poświęcić na daną pozycję całą moc swojego procesora i tyle czasu, ile on uzna za stosowne. Dlatego analizy dokonywane lokalnie są nie tylko wiarygodniejsze, ale i czasem udowadniają, że ocena po stronie serwera była błędna. Jeśli tworzymy aplikację, która ocenia pozycję po stronie serwera musimy się z tym pogodzić - nasze analizy i tak będą lepsze niż „ludzkie przemyślenia” i choć w czysto teoretycznym sensie nie będą one wiążące, to w praktyce będą miały jak najbardziej rzeczywiste zastosowanie. Jakie są zatem zalety uruchamiania silnika szachowego po stronie serwera?

- Obliczenia dostępne są zazwyczaj szybciej, niż w przypadku samodzielnego obsługiwanego programu grającego (zwłaszcza jeśli śledzi się więcej jak jedną partię)
- Dużo łatwiej jest zarządzać obserwowaniem analiz
- Unika się błędów związanych ze złym wprowadzeniem posunięć (każdy krok pomiędzy

²⁵W ramach obliczeń tworzone jest drzewo o bardzo dużej arności. Wówczas nawet drzewo o wysokości kilku półposunięć może zawierać dziesiątki tysięcy liści

ChessBomb don't Chess

Rooms | London Classic 2010 | ANAND - KRAMNIK | NAKAMURA - ADAMS | HOWELL - MCSHANE | **Reggio Emilia 2010** | Exit | Login

Room Info

Standings

#	Name	ELO	Pts	S-B
1	Gashimov Vugar	2733	6	26.5
	Vallejo Pons Francisco	2698	6	23.75
3	Movsesian Sergei	2721	5	21.75
	Navara David	2708	5	19.25
	Ivanchuk Vassily	2764	5	19
6	Caruana Fabiano	2709	4.5	19.5
	Onischuk Alexander	2683	4.5	19.25
8	Morozevich Alexander	2700	4	17.25
9	Short Nigel	2680	3.5	13.75

Chat

pregnator: when will round 9 start?

Bacchante: lol it just writes when it will start in the first line above your post and you ask when its gonna start.)

siamesedream: Vugar won tournament?

Zeblakob80: when next game to start?

JKnight: Never

JKnight: It's over

Zeblakob80: thanks !!!

km_nikolaev: so vugar won it, right?

cafe9stream: yes, in spite of the tie with Vallejo Pons Francisco

siamesedream: 'Gashimov won the title on tiebreaks over Vallejo Pons'

siamesedream: http://2.bp.blogspot.com/_Q0ITPVTG40/TSZ_BdW4l0U/AAAAAAAAABk6o/GTmWnXyLW/s1600/Players%252BStaff.JPG

Bacchante: lol @ chucky

Only registered members can post

[Help ?](#)

Games

Round	White	ELO (W)	Black	ELO (B)	Result
▼ 09	Godena Michele		Caruana Fabiano		½-½
	Ivanchuk Vassily		Morozevich Alexander		1-0
	Movsesian Sergei		Vallejo Pons Francisco		½-½
	Navara David		Short Nigel		1-0
	Onischuk Alexander		Gashimov Vugar		½-½
► 08					
► 07					
▼ 06	Caruana Fabiano		Vallejo Pons Francisco		½-½
	Gashimov Vugar		Ivanchuk Vassily		1-0
	Morozevich Alexander		Onischuk Alexander		0-1
	Navara David		Godena Michele		1-0
	Short Nigel		Movsesian Sergei		0-1
► 05					
▼ 04	Gashimov Vugar		Godena Michele		1-0
	Ivanchuk Vassily		Movsesian Sergei		½-½
	Morozevich Alexander		Caruana Fabiano		½-½
	Navara David		Onischuk Alexander		½-½
	Short Nigel		Vallejo Pons Francisco		0-1
► 03					
► 02					
► 01					

Connected | [Help ?](#)

© 2008-2010 ChessBomb and Chessdom. All rights reserved. | [Contact](#) | [Premium](#) | [Blog](#) | [Embed](#) | [Archive](#) | [Links](#) | [Privacy](#)

Rysunek 4: Serwis ChessBomb - widok na listę partii

wykonaniem posunięcia przez szachistę a wyświetleniem być może na drugim krańcu świata analizy tej pozycji jest zautomatyzowany)

- Można korzystać z analiz, nawet jeśli nie posiada się programu grającego

Jako pierwsze rozwiązanie tego typu ChessBomb nie jest pozbawiony pewnych wad, jednak testy przeprowadzone z użytkownikami wykazały, iż większą część czynności wykonuje się na nim w sposób intuicyjny i nie przysparzający problemów. Serwis posiada dość niewielką liczbę funkcjonalności, jednak te, które są dostarczane, są relatywnie łatwe w użyciu nawet dla niedoświadczonego użytkownika. Takie wnioski wyniknęły zarówno z ankiet, jak i z testów z użytkownikami.

Wygląd interfejsu programu ChessBomb często uznawany jest za nieciekawym, tym bardziej że twórcy nie udostępnili żadnego sposobu jego zmiany. Za bardzo wygodny w użytkowaniu uznany jest system zakładek, za pomocą których można bez najmniejszych problemów przełączać się między oknami poszczególnych partii.

3.5 Cechy wspólne dotychczasowych rozwiązań oraz ich ocena

Istnieją elementy, które są wspólne dla dużej liczby istniejących produktów.

1. **Wyświetlanie nazw linii i kolumn** Jedną z cech wspólnych dla większości programów jest wyświetlanie nazw linii i kolumn. Przypomina ono rozwiązanie jakie występuje na zwyczajnych, papierowych lub drewnianych szachownicach. Twórcy interfejsu podświadomie stosują się do zasady, aby czerpać wzorce ze świata fizycznego. W tej kwestii pozostaje jednak jeszcze jedno pytanie - czy należy opisywać linie i przekątne z jednej czy z obu stron. W dotychczasowych rozwiązaniach oba pomysły miały dotychczas zbliżoną popularność. Moim zdaniem trzeba to dostosować już do samego interfejsu - ile mamy miejsca na dodatkowe oznaczenie szachownicy.
2. **Przedstawianie czasu** Podobnie sprawa się ma z przedstawianiem czasu. To dotyczy zarówno transmisji na żywo (gdzie podaje się czas, jaki pozostał zawodnikom), jak i partii rozgrywanych między internautami na kafejkach szachowych (gdzie pokazywany jest czas użytkowników). Wszędzie korzysta się ze standardu pokazywania, jak wiele czasu zostało zawodnikom. Jest to naturalne podejście (znow - takie samo jest w przypadku fizycznych zegarów szachowych) i każdy interfejs realizuje go w podobny sposób²⁶. W tym przypadku można rozróżnić transmisje i grę na kafejkach, gdyż w transmisjach samo opóźnienie może być kilkudziesięciosekundowe²⁷, w kafejkach internetowych zaś nawet półsekundowe opóźnienie może być kluczowe²⁸. Dlatego też w kafejkach często podaje się czas z dokładnością do dziesiątych części sekundy, zaś przy transmisjach dokładność tego stopnia jest absolutnie niepotrzebna.

²⁶Nigdy nie podaje się czasu zużytego, gdyż w przypadku zakończenia partii nie ma on wpływu na żaden aspekt wyniku

²⁷Z powodów podanych w rozdziale dotyczących szachownic elektronicznych

²⁸Wiele osób, mnie włączywszy, grywa partie, w których na całość przeznaczona jest zaledwie jedna minuta

4 Projektowanie własnego interfejsu

Nadszedł czas, aby przejść od etapu oceniania gotowych rozwiązań do zaproponowania własnych. Moim założeniem jest, iż nie będę starał się wymyślić czegoś absolutnie nowatorskiego, a postaram się ulepszyć rozwiązania stosowane w serwisie ChessBomb. Zamierzam korzystać, podobnie jak ów serwis, z systemu zakładek reprezentujących turniej albo konkretną partię. W odróżnieniu od stosowanych w ChessBombie rozwiązań pozwolę użytkownikom na utworzenie kilku partii w jednej łączonej zakładce. Do uzupełnienia w zależności od tego, jak się uda

4.1 Rodzaje użytkowników

Nie sposób myśleć o programie jako o stuprocentowo automatycznej witrynie. Niektóre czynności z konieczności będą musiały być wykonywane przez administratora strony. Oczywiście należy dążyć do zminimalizowania nakładu pracy ludzkiej poprzez maksymalne ułatwienie tych czynności, jednak najprostsze operacje będą musiały być wykonywane przez administratora strony. Ogólnie planuję podzielić użytkowników na następujące kategorie:

- Administratorzy strony
- Moderatorzy treści
- Zarejestrowani użytkownicy strony
- Niezarejestrowani użytkownicy strony

Jest to dość typowy podział wśród witryn internetowych i będzie miał zastosowanie także przy moim projekcie. Zadanie administratora będą w zasadzie jedno - dodawanie turniejów. Aby móc to uczynić będzie udostępnione dla niego specjalne okno, w którym definiować będzie się wszystkie dane dotyczące danych zawodów. Oczywiście administrator będzie miał wszystkie uprawnienia użytkowników pozostałych kategorii.

Moderator strony będzie miał możliwość zarządzania użytkownikami witryny - będzie mógł on usuwać ich wpisy w oknie rozmów, będzie mógł także blokować czy też kasować konto użytkownika. Nie będą oni mieć dostępu do żadnych kluczowych dla wyświetlania witryny funkcji, będą jednak pełnić ważną rolę w momencie rozwinięcia systemu komentowania na bieżąco trwających partii.

Zarejestrowany użytkownik strony będzie miał możliwość korzystania z głównych funkcji strony - oglądania i komentowania partii. Odróżnia się go od niezalogowanego użytkownika, który nie ma możliwości wypowiadania się na stronie. Zalogowany użytkownik będzie także posiadał opcje edycji swoich danych osobowych, a także wysyłania wiadomości do innych użytkowników. Poprzez specjalny panel będzie mógł także skontaktować się z administratorami i moderatorami strony.

4.2 Przypadki i scenariusze użycia oraz papierowy interfejs

4.2.1 Scenariusze użycia

Scenariusze użycia są przydatnymi narzędziami podczas projektowania interfejsów. Pomagają one wyobrazić sobie kontekst, w jakim będzie używana aplikacja. Dzięki temu można lepiej zrealizować funkcje, tak aby przykładowym użytkownikom dobrze się z nich korzystało.

Przykładowe scenariusze użycia

Scenariusz #1

John Doe siedzi w swoim pokoju o godzinie 8:30 rano w niedzielę. Normalnie jeszcze by spał, ale jest relacja z Bardzo Ważnych Zawodów. Uruchomił stronę www, która to uruchomiła witrynę, na której prowadzona jest relacja na żywo. Na stronie głównej zauważył informację, że tego dnia nie będzie komentarza ludzkiego, jedynie automatyczny. John Doe uruchamia witrynę, która wyświetla mu listę wszystkich turniejów dostarczanych przez stronę internetową. Lista jest posortowana chronologicznie, więc wybiera z niej Bardzo Ważne Zawody. Pokazuje mu się lista aktualnie granych partii. Lista jest posortowana względem szachownic, jednak John wybiera opcję zmiany na sortowanie alfabetyczne. Wybiera z listy partię swojego ulubieńca, Seana Smitha, który gra z Andriejem Kałmukowem. W wyniku wybrania otwiera się nowa zakładka z partią Smith-Kałmukow. John Doe widzi, iż zostało wykonanych już 15 posunięć i białe zużyły 20 minut, czarne zaś 10. Z przebiegu partii wybiera pierwsze posunięcie i wybierając opcję pokaż następny ruch przechodzi do końca partii. Widzi ocenę pozycji po najlepszym ruchu Smitha i dwukrotnie klikając w wariant wyświetla go w postaci graficznej.

Scenariusz #2

Michael Collins siada wieczorem do komputera, aby zobaczyć wyniki z dzisiejszej rundy Bardzo Ważnego Turnieju. Uruchamia stronę internetową, po czym wchodzi do opcji logowania. Wpisuje swoje dane i bezproblemowo loguje się na serwer. Witryna wyświetla mu listę turniejów, po czym Michael wybiera Bardzo Ważny Turniej. Wyświetliła mu się lista partii z wpisanymi do niej wynikami. Michael otwiera zakładki z partiami z sześciu pierwszych szachownic i przegląda każdą z nich osobno. Po przejrzaniu partii zamyka zakładkę. Po przejrzaniu wszystkich partii pozostaje w zakładce turnieju i w „shoutboxie” pisze komentarz na temat rundy. Po wykonaniu tej czynności wylogowuje się i zamyka przeglądarkę.

Scenariusz #3

Jack Wilshire przychodzi do komputera około godziny 12. Nie włącza niczego, gdyż zostawił przeglądarkę włączoną poprzedniej nocy. Widzi, że może kliknąć na pole oznaczające nową rundę i wyświetli mu się lista partii. Niektóre partie się zakończyły i są wpisane ich wyniki. Wchodzi na partię z pierwszej szachownicy i widzi, że poza posunięciami jest w zapisie partii komentarz słowny. Jack odchodzi od komputera i wraca po kwadransie z gorącą kawą. Widzi natychmiast, że zostało zagranych kilka ruchów, bo mimo iż nie dotknął ani

myszki, ani klawiatury, pozycja się zmieniała. Po kolejnych pięciu minutach widzi wpisany wynik partii – zakończyła się remisem.

4.2.2 Przypadki użycia

Przypadki użycia ukazują interakcję między użytkownikiem a systemem, pokazując ścieżki za pomocą których użytkownik może osiągnąć wyznaczone cele i sposób, w jaki system realizuje wszystkie kroki pośrednie. Są one szczególnie przydatne podczas implementacji i późniejszej ewaluacji interfejsu - można porównać przypadki użycia z fantycznym sposobem dochodzenia do celów w naszym programie.

4.2.3 Papierowy prototyp

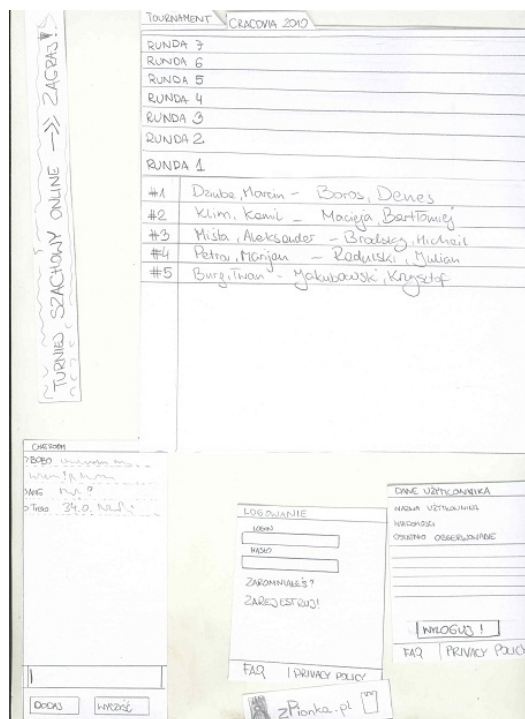
Papierowy prototyp, jak sama nazwa wskazuje, jest szablonem wyglądu interfejsu stworzonym za pomocą przyrządów biurowych. Mimo iż wydaje się to nieco nieprofesjonalnym podejściem, pozwala on na szybkie i mało kosztowne przetestowanie użyteczności interfejsu. Tworząc taki schemat twórca zupełnie odcina się od sposobu w jaki tworzy interfejs i skupia się na samych funkcjonalnościach. W wyniku temu można szybko znaleźć największe problemy dotyczące użyteczności programu.

Papierowy prototyp pokazuje wszystkie najważniejsze elementy interfejsu, pozwalając testowanej osobie zobaczyć, jaki układ informacji będzie prezentowany na stronie. Uczestnicy testów dostali listę zadań do wykonania, podczas gdy obserwoałem sposób i mierzyłem czas jaki zajmuje im dojście do określonego celu.

Lista i kolejność zadań była następująca:

1. Zalogować się na stronie, a później się wylogować.
2. Zalogować się na stronie, znaleźć listę turniejów, wybrać z niej najstarszy turniej i napisać komentarz na chatroomie.
3. Pozostając w tym turnieju znaleźć partię z pierwszej rundy graną na pierwszej szachownicy
4. Przewinąć partię do początku, a następnie ruch po ruchu ją przejrzeć

Uczestnicy testów byli zachęceni do wygłaszania swoich uwag i propozycji, a każde badanie było nagrywane, by można było odtworzyć przebieg badania. Nie było także ustalonego limitu czasu na wykonanie danej czynności. Jako grupę testową użyłem szachistów w wieku 15-40 lat - docelową grupę użytkowników programu. Byli wśród nich zarówno doświadczeni użytkownicy interfejsów szachowych, jak iniedoświadczeni szachiści



Rysunek 5: Papierowy prototyp interfejsu

4.2.4 Wnioski z testów na papierowym prototypie oraz ich wpływ na kształt interfejsu

4.3 Wygląd interfejsu

Papierowy prototyp nie testował ustaleń dotyczących niektórych aspektów grafiki przyszłej aplikacji. Powód tego jest prosty - kolory ze skali RGB czasem nie tak łatwo reprezentować za pomocą fizycznych przyrządów. I choć najlepiej pozwolić użytkownikom na wybór preferencji graficznych, trzeba wybrać kolory choćby dla dobra pierwszego wrażenia przyszłego użytkownika.

4.4 Dobór kolorów interfejsu

4.5 Rodzaje figur i szachownic

5 Impentacja programu

5.1 Technologia

5.1.1 Ruby o Rails

“Ruby on Rails jest przełomem w dziedzinie programowania aplikacji internetowych. Potężne aplikacje, których tworzenie do tej pory zabierało tygodnie czy miesiące, są teraz tworzone dosłownie w kilka dni.” -Tim O'Reilly, Założyciel O'Reilly Media²⁹

O produkcie często najlepiej świadczy renoma ich klientów. Duże międzynarodowe firmy nie korzystają ze złych rozwiązań ani w biznesie, ani w procesie tworzenia aplikacji. O sile frameworka Ruby on Rails świadczą między innymi tak potężne serwisy internetowe jak Twitter czy Gruopon. Język Ruby powstał w roku 1995, stworzony przez Yukihiro "Matza" Matsumoto. Jest w pełni obiektywnym i typowanym dynamicznie językiem bazującym na wielu językach, takich jak CLU, Eiffel, Lisp, Perl, Python czy Smalltalk Ruby cechuje się dużą zwieżnością kodu, przy posiadaniu wciąż czytelnej składni. Co ciekawe, w Ruby można modyfikować każdą klasę, włączywszy w to nawet klasę string. Jedną z głównych zalet języka jest jednak framework Ruby on Rails, który jest świetnym narzędziem do tworzenia aplikacji webowych

Ruby on Rails powstał jako niezależny projekt stworzony głównie przez duńskiego programistę Davida Heinemeiera Hanssona. Ruby on Rails jest narzędziem do szybkiego tworzenia aplikacji webowych **opartych na architekturze Model-View-Controller** przy zachowaniu zasady DRY³⁰ oraz reguły Convention over Configuration³¹. Jedną z większych zalet Ruby on Rails jest fakt, że posiada bardzo użyteczny moduł ActiveRecord oparty na ORM³², który pozwala na tworzenie modeli w architekturze MVC niezależnych od faktycznej bazy danych. Kolejną zaletą tego frameworku jest dostępność świeżo rozwijanych i użytecznych bibliotek. Jest też, co warto podkreślić, bardzo prosta w nauce - sam nie znałem tej technologii w momencie rozpoczęcia pisania pracy magisterskiej.

5.1.2 JRuby on Rails

Choć Ruby jest już bardzo rozwiniętym językiem, czasami przydaje się jednak możliwość wykorzystania kodu napisanego w innych językach programowania. Takim rozwiązaniem jest JRuby, który pozwala na uruchomienie kodu Ruby na JVM (wirtualna maszyna Javy) oraz na korzystanie z klas napisanych w Javie. Podobnie jest samo Ruby on Rails został

²⁹O'Reilly Media to uznana amerykańska kompania medialna

³⁰Don't Repeat Yourself - unikaj powtarzania kodu

³¹Minimalna początkowa konfiguracja zastąpiona gotowymi wzorcami

³²Object-Relational mapping

on stworzony przez grupę programistów, którzy chcieli stworzyć jeszcze bardziej użyteczne narzędzie do tworzenia stron webowych. Dzięki bibliotekom JRuby można umieszczać strony napisane we frameworku Rails na urządzeniach przenośnych. Problemem z JRuby jest jedynie lekkie opóźnienie do klasycznego Ruby, najnowsze funkcje z frameworka Rails nie zawsze od razu działają pod JRuby.

5.1.3 Dodatkowe biblioteki

W programie wykorzystywałem gotowe biblioteki. Główną z nich jest Chesspresso, biblioteka napisana w Javie na licencji LGPL, służąca do parsowania i interpretowania plików szachowych o rozszerzeniu pgn, a także interpretowania i tworzenia notacji FEN. Kolejną biblioteką jest JavaScriptowa jChess oprta na znanej bibliotece jQuery, która wizualnie prezentuje partie szachową. Kolejną biblioteką jest paperclip - plugin w Rubym do importowania zdjęć, który można łatwo zintegrować z wymienionym przed chwilą jQuery.

5.2 Struktura programu

Ruby on Rails odgórnie narzuca architekturę MVC i choć oczywiście istnieją sposoby, aby korzystać z innych architektur, całe wsparcie frameworka najskuteczniej działa dla tego właśnie schematu. Ruby on Rails domyślnie interpretuje, że dla każdego modelu istnieje jeden kontroler, natomiast widoki są ściśle powiązane z metodami kontrolera dzieląc takie same nazwy. Pliki modeli i kontrolerów są pisane w Ruby, natomiast pliki widoków w specjalnych plikach html.erb, które są połączeniem pliku html z kodem Ruby umieszczanym między odpowiednimi znacznikami. Przy żądaniu wyświetlenia pliku html najpierw uruchamiany jest kontroler, który komunikując się z modelem zwraca do widoku potrzebne wartości.

5.2.1 Funkcjonalności

Jak zdecydowana większość programów sieciowych głównym odbiorcą aplikacji będzie zwyczajny użytkownik, wydzielona część z nich będzie mogła jednak wykonywać nieznacznie więcej czynności. Uprawnienia w aplikacji działają hierarchicznie, czyli administrator może zrobić wszystko to samo co zwyczajny użytkownik, ale będzie miał dodatkowe funkcje dostępne

Użytkownik

1. Założenie konta
2. Zalogowanie się na konto
3. Wylogowywanie się z konta
4. Edycja danych konta
5. Wysyłanie i odbieranie prywatnych wiadomości od innych użytkowników

6. Komentowanie artykułów
7. Usuwanie własnych komentarzy
8. Przeglądanie listy turniejów, rund, partii
9. Przeglądanie partii zawodników
10. Dodawanie partii do listy obserwowanych oraz usuwanie z tej listy
11. Przeglądanie pojedynczej partii w oknie graficznym
12. Przeglądanie zapisu partii i nawigacja za jego pomocą do odpowiednich miejsc w rozgrywce
13. Przeglądanie odpowiednich wariantów dla obecnej pozycji
14. Dodawanie wiadomości dotyczących turniejów

Moderator

1. Dodawanie nowych partii, rund i turniejów
2. Usuwanie komentarzy innych użytkowników
3. Dodawania i usuwanie informacji z panelu komentatorskiego turnieju
4. Dodawanie i usuwanie artykułów

Administrator

1. Usuwanie innych użytkowników
2. Zmiana uprawnień innych użytkowników

5.2.2 Struktura bazy danych

Podstawowymi tabelami, w których będzie przechowywana największa liczba informacji będą te zawierające dane dotyczące różnych turniejów. Wyróżniam wśród nich cztery najważniejsze tabele - Turniej, Runda, Partia i Pozycja, gdzie na każdym kroku nie tylko występuje relacja "jeden do wielu", ale także te proporcje będą w miarę stałe. Stosunki między liczbą poszczególnych krotek można oszacować następującą równością $\text{Turniej} = 9 * \text{Runda} = 10 * 9 = 90$ $\text{Partia} = 10 * 9 * 40 = 3600$ Pozycja . Ogólny schemat bazy danych przedstawia następujący schemat. Większość pozostałych tabel służy do zarządzania użytkownikami, ich statusem, wysłanymi wiadomościami, komentarzami oraz innymi funkcjonalnościami dostępnymi dla użytkownika.

5.3 Fragmenty najciekawsze algorytmicznie

6 Ewaluacja programu

6.1 Wyniki ewaluacji programu

6.2 Wnioski i ulepszenia wynikające z ewaluacji

6.3 Raport z produktu końcowego

7 Podsumowanie