

Zakres egzaminu dyplomowego

1. Architektura przetwarzania informacji i standardy komunikacji w systemach Internetu rzeczy (ang. Internet of Things - IoT).

Internet Rzeczy (Internet of Things - IoT) to koncepcja powszechnie przedstawiana jako kolejny etap komputerowej i sieciowej rewolucji, w kierunku rozproszenia modułów sieciowych i samych informacji, gdzie każdy obiekt w świecie rzeczywistym może automatycznie połączyć się z siecią i komunikować się z dowolnym innym modułem do niej podłączonym. IoT obejmuje obecnie szereg technologii i obszarów badawczych, które mają na celu wykorzystanie i rozszerzenie istniejącej sieci Internet, jako platformy komunikacyjnej również dla różnego typu obiektów, urządzeń i modułów występujących w otoczeniu człowieka.

STANDARDY:

Wi-Fi

- Sieci lokalne LAN (local area network)
- Zasięg od kilku metrów do kilku kilometrów
- Przepustowość do 900 Mb/s
- Pracują w częstotliwości 2,4 GHz i 5 GHz (5 Ghz zapewnia szybszą transmisję danych)
- Oparta na topologii gwiazdy
- Moc wystarcza na pokrycie zasięgiem mieszkania / domu
- Do pojedynczej sieci można podłączyć ok 250 urządzeń (zależy od punktu dostępowego / routera)
- Powszechność standardu
- Duże zużycie zasobów pamięci, energii

Bluetooth

- Bluetooth classic, później opracowany Bluetooth Low Energy (BLE)
- Dla IoT lepszy BLE przez niski pobór mocy
- Pracują w paśmie 2,4 GHz

- W BLE prędkość 2 Mb/s
- Oparta na topologii gwiazdy
- Z punktu widzenia producentów urządzeń IoT atutem jest szeroka dostępność infrastruktury
- Wadą jest umiarkowany zasięg oraz ograniczona liczba urządzeń, które mogą być jednocześnie podłączone do jednej sieci.
- Bluetooth Mesh Networking pozwala na topologie siatki i większy zasięg

ZigBee

- Pasmo 2,4 Ghz
- Topologia siatki
- Maksymalny zasięg bezpośredniej komunikacji pomiędzy dwoma węzłami sieci wynosi ok. 100 m
- Wysoka interoperacyjność pomiędzy modułami różnych producentów
- Niski pobór pamięci, ok. 4-krotnie mniejszy niż bluetooth
- Maksymalna szybkość transmisji danych wynosi 250 kbps
- Często wykorzystuje się w wielu rozwiązaniach z zakresu inteligentnego budynku
- ZigBee Green Power, zoptymalizowany pod kątem bardzo niskiego zużycia energii

6LoWPAN (IPv6 over Low Power Wireless Personal Area Networks)

- Kłopoty z interoperacyjnością modułów pochodzących od różnych producentów
- Topologia siatki
- Pasmo 2,4 GHz
- Szybkość transmisji danych do 250 kbps
- Energooszczędny standard

Thread

- Jedna z najbardziej popularnych implementacji standardu 6LoWPAN
- Cel - wysoka interoperacyjność modułów różnych producentów, czyli eliminacja jednej z głównych wad standardu 6LoWPAN
- Maksymalna prędkość 250 kbps
- Maksymalny rozmiar 250 węzłów

WirelessHART

- Pasmo 2,4 GHz
- Wykorzystywany głównie w aplikacjach przemysłowych
- Wysoka odporność na interferencje elektromagnetyczne
- Szeroko stosowany w automatyce
- Niewielkie czasy opóźnień transmisji
- Maksymalny zasięg transmisji nie przekracza 100 m
- Sieć może zostać skonfigurowana zarówno w topologii gwiazdy, jak i siatki

LoRaWAN

- Sieć typu LPWAN (Low Power Wide Area Network)
- Topologia gwiazdy
- Pracuje na częstotliwościach poniżej 1 GHz, dzięki czemu charakteryzuje się dużym zasięgiem (kosztom niższej szybkości transmisji niż w przypadku pasma 2,4 GHz)
- Maksymalny zasięg stacji bazowej wynosi od 15 km w terenie niezabudowanym do ok. 2 km w terenie silnie zurbanizowanym
- Jedna stacja bazowa może obsługiwać do 20 tysięcy urządzeń końcowych
- Szybkość transmisji wynosi od 0,3 do 50 kbps
- Zaletą jest duży zasięg i wysoka energooszczędność
- Możliwość precyzyjnego zarządzania czasem pracy węzłów sieci - przez większość czasu mogą pozostawać nieaktywne, wybudzając się tylko na czas transmisji oraz nasłuchiwanie odbioru

Sigfox

- Pasma poniżej 1 GHz
- Przeznaczone dla modułów, które wymagają przesyłania niewielkiej ilości informacji z niezbyt dużą częstotliwością
- Topologia gwiazdy, której centralnym elementem jest stacja bazowa
- Maksymalny zasięg stacji wynosi do 30 km w terenie niezabudowanym
- Szybkość transmisji do 100 bps
- Przeznaczona jest przede wszystkim dla aplikacji, które komunikują się rzadko oraz bardzo niewielką ilością danych, jak np. rozproszone systemy pomiarowe do pomiaru zużycia wody lub energii elektrycznej

Sigfox - wykorzystuje technologię UMB (Ultra Narrow Band), co sprawia, że nadaje się do transmisji jeszcze mniejszych pakietów danych z prędkością od 10 b/s do 1000 b/s. Typowy pobór prądu przez modem Sigfox w trybie aktywnym wynosi od 20 do 70 mA, a trybie uśpienia jest praktycznie pomijalnie mały. Tak nieduży pobór prądu sprawia, że urządzenie brzegowe może pracować przez całe lata na zasilaniu baterijnym, zwłaszcza przy okazjonalnym zapotrzebowaniu na dane. Aktualnie jednak zasięg tych sieci jest ograniczony w znacznej mierze do sieci Sigfox wdrożonych np. w dużych miastach w Europie Zachodniej, przeważnie w krajach Beneluksu, Francji, Portugalii, Hiszpanii i Wielkiej Brytanii.

Narrowband-IoT (NB-IoT) - nowo zatwierdzony standard komunikacji komórkowej, którego głównym celem jest sprostanie wszystkim oczekiwaniom i wymogom Internetu Rzeczy. Standard NB-IoT został zatwierdzony w czerwcu 2016, zaledwie po 9 miesiącach pracy, przez stowarzyszenie 3GPP, które zrzesza najbardziej liczące się organizacje zajmujące się tworzeniem rozwiązań telekomunikacyjnych. Duże tempo prac wskazuje istnienie ogromnego popytu na tego typu usługę. Pełna nazwa standardu

to LTE Cat NB1 Release 13. Łączy on wszystkie cechy, które są utożsamiane z ideą Internetu Rzeczy, a więc:

- **wąskie pasmo radiowe**, co wskazuje na zmniejszenie prędkości połączenia kosztem większej możliwej liczby jednoczesnych połączeń i zasięgu,
- **wysoka sprawność energetyczna komunikacji**,
- **szeroka dostępność i niski koszt modułów**.

LTE kategorii M (LTE-M) - LTE-M i NB-IoT, wzajemnie się uzupełniają i odpowiadają na złożoność wymagań i scenariuszy wdrożeń przypadków użycia LPWA (*Low Power Wide Area*) IoT jak najlepiej wykorzystując istniejące sieci. Interoperacyjność jest bardzo ważna w IoT – na przykład, 5G to nowa technologia, która łączy nowe i istniejące technologie dostępu radiowego i oprócz polepszania szerokopasmowych usług mobilnych odpowiada na potrzeby rosnącej gamy przypadków użycia IoT. Technologia LTE już teraz jest szczególnie dobrze przystosowana do wspierania zastosowań o wysokiej wydajności i krótkim czasie oczekiwania. Wymagania niższej mocy, kosztów i skomplikowania towarzyszące NB-IoT i LTE-M doprowadzą do rozszerzenia technologii komórkowych na więcej nowych zastosowań.

[\[http://di.com.pl/standardy-komunikacji-specjalizowane-do-obslugi-iot-56337\]](http://di.com.pl/standardy-komunikacji-specjalizowane-do-obslugi-iot-56337)

Dodatkowo moja prezentacja:

Komunikacja przewodowa i bezprzewodowa w systemach IoT:

<https://drive.google.com/open?id=1Si45dpIIYjhDU5AV5y8arFJmYyHspdUe>

2. Architektura REST (Representational State Transfer). Charakterystyka, właściwości i zastosowania.

Representational State Transfer (REST, ang. zmiana stanu poprzez reprezentacje)

– styl architektury oprogramowania wywiedziony z doświadczeń przy pisaniu specyfikacji protokołu HTTP dla systemów rozproszonych. REST wykorzystuje m.in. jednnorodny interfejs, bezstanową komunikację, zasoby, reprezentacje, hipermedia, HATEOAS.

Zaproponowany przez Roya T. Fieldinga w 2000 roku.

API – Application Programming Interface – zestaw reguł definiujący komunikację pomiędzy programami komputerowymi.

Zasady tworzące REST

Aby API można nazwać RESTful lub API RESTowym musi ono spełniać kilka założeń:

1. **Odseparowanie interfejsu użytkownika od operacji na serwerze.** Klient poprzez “wydawanie poleceń” nie ma wpływu na to co się dzieje po stronie serwera. Działa to również w drugą stronę – serwer daje klientowi jedynie odpowiedź i nie ma prawa ingerować w UI. Pozwala to na korzystanie z jednego REST API w wielu niezależnych od siebie aplikacjach, a dane pozostaną spójne.
2. **Bezstanowość** – mówi się że REST jest stateless – oznacza to, że każde zapytanie od klienta musi zawierać komplet informacji oraz, że serwer nie przechowuje stanu o sesji użytkownika po swojej stronie.
3. **Cacheability** – odpowiedź, którą użytkownik otrzyma z REST API musi jasno definiować, czy ma ona być cacheable czy non-cacheable. Ma to znaczenie przy danych, które bardzo szybko stają się nieaktualne oraz przy danych, które aktualizują się relatywnie rzadko – nie ma sensu na przykład cache’ować współrzędnych geograficznych pędzącego samolotu, natomiast już jego kolor czy nazwę już tak.
4. **Endpointy**, czyli tak naprawdę adresy, które pozwalają na dostęp do informacji powinny jednoznacznie wskazywać do jakiego zasobu się odwołują. Z ich budowy powinniśmy wiedzieć jaki konkretnie zasób otrzymamy. Co ważne – dane otrzymywane z API powinny być niezależne od schematu bazy danych w jakiej są przetwarzane. Oczywiście nie ma przeciwwskazań, aby struktura danych

wyglądała identycznie jak schemat bazy danych – niemniej jednak struktura w żaden sposób nie powinna zależeć od tego schematu.

5. **Separacja warstw** – powinniśmy oddzielić warstwy dostępu do danych, logiki oraz prezentacji. Pozwala to na dowolne operacje na danych – nie wymuszamy na użytkowniku konkretnego działania na nich, ani wyświetlania ich w konkretny sposób. Ponadto pośrednie i zewnętrzne API wykorzystywane przez serwer(!) mogą być wykorzystywane bez wiedzy klienta. Przykładem może być wcześniej wspomniany samolot, gdzie informacja np. o kolorze może pochodzić z zupełnie innego API – klient nie musi o tym wiedzieć.
6. **Możliwość udostępniania apletów i skryptów klientowi** – jest to opcjonalna reguła, aczkolwiek warto rozważyć jej zastosowanie. Jeśli wiemy, że klienci będą wykonywać konkretne operacje na konkretnych danych możemy im udostępnić gotowe do tego rozwiązania

Podstawowe metody:

- **GET** – służy do pobierania danych. Tutaj wystarczy podać odpowiedni endpoint, ewentualnie zmodyfikować nagłówki(headers) zapytania.
- **POST** – służy tworzeniu i przesłaniu nowych danych. W tym przypadku konieczne jest już stworzenie ciała(body), w którym prześlemy dane do naszego REST API.
- **PUT** – również służy przesyłaniu danych, lecz najczęściej w celu aktualizacji tych danych. Tutaj również wymagane jest przesłanie danych w ciele
- **DELETE** – metoda służąca do usuwania danych. W tym momencie chcę wspomnieć o technice tak zwanego soft-delete. Mówiąc skrótem polega to na tym, że kasując dane za pomocą API tak naprawdę tylko dodajemy do encji informację o tym, że została ona usunięta. W rezultacie dane pozostają nadal w bazie, lecz nie są one dostępne z poziomu API. Mechanizm ten należy już zaimplementować w samym API i nie ma on nic wspólnego z HTTP.
- **PATCH** - PUT służy do aktualizacji/zapisu zasobu, ale tylko w sytuacji, gdy klient przyśle cały zasób. Bardzo często gdy mamy do czynienia z dużymi zasobami i chcemy oszczędzić transferu danych interesuje nas aktualizacja tylko jednego pola. Możliwości są dwie. Pierwszą jest użycie metody PATCH. Zgodnie ze specyfikacją, jako body należy podać opis zmian dla danego zasobu.

Zalety:

- Największą z zalet jest przede wszystkim uniwersalność. Załóżmy, że potrzebujesz stworzyć aplikację na telefon i stronę internetową dla księgarni. Możesz stworzyć jedno API, z którego będzie korzystała zarówno aplikacja jak i strona internetowa!
- Po drugie – jeśli napisany interfejs programistyczny jest zgodny z zasadami REST, to w trakcie procesu pisania aplikacji będziesz intuicyjnie tworzył endpointy i dzięki nim wygodnie operował na danych.
- Dane jakie otrzymujemy z API, najczęściej są w wygodnym do pracy formacie – JSON, czyli JavaScript Object Notation. Czasami zdarzy się jeszcze format XML, niemniej jednak spotykam się z nim coraz rzadziej.
- Możemy pozyskiwać dane jednocześnie z wielu źródeł. Na przykład możemy stworzyć aplikację wykorzystującą logowanie za pomocą Facebooka oraz mapy Google. W obu przypadkach skorzystamy z API oferowanych przez te firmy.
- Kolejną z zalet jest możliwość odseparowania warstwy klienta, od warstwy serwerowej. Alternatywą dla API jest tworzenie widoków za pomocą silników szablonów takich jak Pug, Twig, czy Blade.

3. Cele, konstrukcja i zastosowania Internetu rzeczy. Potencjalne korzyści i zagrożenia.

Internet rzeczy (ang. Internet of Things) jest to koncepcja, wedle której jednoznacznie identyfikowalne przedmioty są w stanie pośrednio lub bezpośrednio gromadzić, przetwarzać lub wymieniać dane za pośrednictwem sieci komputerowej.

Cel:

Podstawowym celem Internetu rzeczy jest stworzenie inteligentnych przestrzeni, tj. inteligentnych miast, transportu, produktów, budynków, systemów energetycznych, systemów zdrowia czy systemów związanych z życiem codziennym. Podstawą rozwoju inteligentnych przestrzeni jest dostarczenie technologii, która zapewni ich realizację.

https://pl.wikipedia.org/wiki/Internet_rzeczy

Konstrukcja:

Wdrożenie Internetu rzeczy wymaga modyfikacji i rozwoju poszczególnych warstw poczynając od warstwy fizycznej, tj. doposażenia w sensory i urządzenia wykonawcze otaczającego nas świata, rozwoju technologii sieciowej, która zapewni łączność między poszczególnymi urządzeniami, a także platformy serwisowej, która dostarczy użytkownikowi interfejsu do korzystania z nowych możliwości.

Warstwa percepcji odpowiedzialna jest za zbieranie danych ze świata rzeczywistego, obejmuje przede wszystkim sensory i elektroniczne identyfikatory. Sensory od lat są używane w przemyśle, zakładach produkcyjnych, samochodach czy służbie zdrowia. Od jakiegoś czasu są na tyle małe i tanie, iż mogą być dodawane do dowolnych urządzeń. Większość współczesnych smartfonów jest wyposażona w czujnik oświetlenia, akcelerometr, magnetometr, żyroskop, czujnik zbliżeniowy, czujnik temperatury, a nawet czujnik wilgotności czy barometr. Kolejna klasa sensorów jest używana w zegarkach, opaskach na nadgarstek, szklach kontaktowych czy materiałach.



Warstwa

urządzeń

Elementy w warstwie urządzeń obejmują fizyczne czujniki i siłowniki, które są podłączone do urządzeń IoT. Same czujniki i elementy takie jak np. siłowniki nie są uważane za "inteligentną" część systemu, ale często łączą się one bezpośrednio lub bezprzewodowo (za pomocą technologii takich jak Bluetooth LE lub ZigBee) z urządzeniami IoT, które mają więcej możliwości przetwarzania.

Niektóre urządzenia IoT komunikują się bezpośrednio z powiązаныmi usługami i aplikacjami w chmurze jednak bardziej powszechnym rozwiązaniem jest to, że urządzenia IoT komunikują się za pośrednictwem bramek, które są urządzeniami pośrednimi. Urządzenia te mają nieco większą moc przetwarzania niż podstawowe elementy w tej warstwie. Urządzenia bramkowe zazwyczaj nie posiadają bezpośrednio podłączonych czujników, ale odgrywają one ważną rolę w procesie gromadzenia danych. Mogą wykonywać podstawowe konwersję analogowo-cyfrowe, skalowanie i inne normalizacje odczytów z surowych danych z czujników.

Warstwa

Edge

Warstwa Edge dotyczy usług analitycznych i przetwarzania wstępnego, które znajdują się na obrzeżach sieci. Analiza danych w warstwie krawędziowej odbywa się w czasie rzeczywistym (lub w czasie zbliżonym do rzeczywistego). Dane do analizy pochodzą z czujników z warstwy urządzeń i są przetwarzane w tym samym punkcie gdzie zostały zebrane. Podstawowe zadania przetwarzania wstępnego, takie jak filtrowanie i agregacja danych, są wykonywane na krawędzi, a następnie tak wstępnie przetworzone dane są przekazywane do usług i aplikacji w chmurze. Dane te służą do dalszego przetwarzania i analizowania.

Warstwa

chmurowa

Po zebraniu danych są one wysyłane do dalszego przetwarzania, przechowywania i używania w aplikacjach w chmurowych. Aplikacje w chmurze przetwarzające dane są często rozszerzane przez klienckie aplikacje mobilne i internetowe, które prezentują dane użytkownikom końcowym i zapewniają dostęp do narzędzi analitycznych. Narzędzia te służą do dalszych poszukiwań i analiz za pośrednictwem interfejsów, pulpitów i wizualizacji.

Zastosowania:

- **SMART HOME**

Fraza „Smart Home” jest najczęściej wyszukiwaną frazą związaną z Internetem rzeczy w Google'u. Za pomocą zestawu czujników i detektorów znajdujących się w mieszkaniu inteligentny dom jest w stanie zarządzać takimi elementami jak:

- oświetlenie
- ogrzewanie
- wentylacja
- system alarmowy
- wiele innych..

- **MEDYCYNĄ**

Może być wykorzystywany między innymi do:

- pomiarów wskaźników życiowych (bicie serca, ciśnienie krwi, ekspozycja na promieniowanie UV, telemedycyna)
- Inteligentne soczewki (aktualnie wykorzystywane głównie przez diabetyków do pomiaru glukozy we łzach)
- Optymalizacja czasu oraz sposobu snu

- **SMART CITY**

- Pomiary zanieczyszczenia powietrza, natężenia dźwięku
- Kontrola zapalania się lamp ulicznych
- Kontrola sygnalizacji świetlnej
- Monitoring
- Pomiary zużycia wody i gazu
- Czujniki pogodowe

- **INNE**

- Połączone samochody (ang. Connected cars)
- Inteligentne parkingi
- Monitorowanie łańcuchów dostaw
- Inteligentne farmy

Korzyści i zagrożenia

+	-
<ul style="list-style-type: none">• Dane: im więcej informacji, tym łatwiej podjąć właściwą decyzję. Wiedza o tym, co można dostać ze sklepu spożywczego, gdy jesteś na zewnątrz, bez konieczności samodzielnego sprawdzania, nie tylko oszczędza czas, ale jest także wygodna.• Śledzenie: Komputery zachowują ścieżkę zarówno pod względem jakości, jak i żywotności rzeczy w domu. Znajomość daty ważności produktów przed ich konsumowaniem poprawia bezpieczeństwo i jakość życia. Poza tym nigdy nie zabraknie niczego, kiedy będzie to potrzebne w ostatniej chwili.• Czas: Ilość czasu zaoszczędzonego na monitorowaniu i liczba wyjazdów w przeciwnym razie byłyby ogromne.• Pieniądze: aspekt finansowy jest najlepszą zaletą. Technologia ta może zastąpić ludzi, którzy są odpowiedzialni za monitorowanie i utrzymywanie zapasów.	<ul style="list-style-type: none">• Zgodność: Na dzień dzisiejszy nie ma standardu oznaczania i monitorowania za pomocą czujników. Konieczna jest jednolita koncepcja, taka jak USB lub Bluetooth, co nie powinno być trudne.• Złożoność: Istnieje wiele możliwości niepowodzenia w przypadku złożonych systemów. Na przykład zarówno Ty, jak i Twój współmałżonek, możecie otrzymywać wiadomości, że mleko się skończyło i oboje możecie skończyć kupowanie tego samego. To daje podwójną wymaganą ilość. Lub występuje błąd oprogramowania powodujący, że drukarka zamawia atrament kilka razy, gdy wymaga pojedynczego wkładu.• Prywatność / bezpieczeństwo: Prywatność to poważny problem z IoT. Wszystkie dane muszą być zaszyfrowane, aby dane dotyczące twojego statusu finansowego lub ilości konsumowanego mleka nie były powszechnie znane w miejscu pracy lub z przyjaciółmi.• Bezpieczeństwo: Istnieje ryzyko, że oprogramowanie zostanie zhackowane, a dane osobowe niewłaściwie użyte. Możliwości są nieskończone. Twoja zmiana recepty lub włamanie na twoje konto może narazić Cię na ryzyko. Dlatego wszelkie ryzyko związane z bezpieczeństwem staje się obowiązkiem konsumenta.

4. Charakterystyka dobrze zaprojektowanej gry (ang. Gameplay).

https://www.gamasutra.com/view/feature/132341/the_13_basic_principles_of_.php?print=1

<https://www.pluralsight.com/blog/film-games/what-makes-a-great-game-the-key-elements-of-successful-games>

Charakterystyka dobrze zaprojektowanej gry:

1. Kierunek

- a. **Kierunek skupienia** - gracz nigdy nie powinien zgadywać na czym powinien się skupić w danym momencie. Powinno się dopuszczać drugorzędne cele, ale główna ścieżka powinna być jasna
- b. **Przewidywanie** - informuj gracza wcześniej, że coś się zaraz wydarzy, np. gracz słyszy dźwięk pociągu zanim go zobaczy
- c. **Ogłaszaj zmiany** - komunikuj graczowi wszystkie zmiany. Im zmiana występuje rzadziej, tym komunikat powinien być wyraźniejszy, np. pojawienie się komunikatu po zakończeniu misji

2. Zachowanie

- a. **Wiarygodne zdarzenia i zachowania** - każde zdarzenie musi zachodzić zgodnie z logiką i oczekiwaniem gracza, np. mieszkańcy wioski reagują pozytywnie na zabicie smoka
- b. **Nakładanie się zdarzeń** - tracona jest dynamiczność, jeżeli zdarzenia wykonują się jeden po drugim, np. wcześniejsze informacje o przyszłych misjach
- c. **Fizyka** - pierwotna logika gracza operuje w znanych prawach fizyki, powinny stanowić one punkt startowy, np. iskry przy ocieraniu pojazdem o ścianę
- d. **Dźwięk** - powinien być odpowiednio dobrany, np. uwzględnianie odległości gracza od źródła

3. Postęp

- a. **Tempo** - należy mieć na uwadze pożądane poczucie pilności, szybkość występowania zdarzeń, wymagany poziom koncentracji oraz częstotliwość powtarzania wydarzeń, np. stwórz miejsca, gdzie gracz może podziwiać widoki oraz gdzie czuje się klaustrofobicznie i pod presją

4. Środowisko

- a. **Rozmieszczenie** - z namysłem rozmieść elementy na ekranie oraz w świecie gry, np. odpowiednia ilość przestrzeni dla danej liczby przeciwników

5. Fundament

- a. **Gracz** - należy się zastanowić w jaki sposób wygląda interakcja gracza ze wszystkimi elementami gry, czy jest ona zgodna z oczekiwaniami
- b. **Komunikacja** - przemyślany musi być sposób komunikacji różnych informacji graczowi, np. drogowskazy będące elementem środowiska

- c. **Atrakcyjność** - czy dany element gry przyciąga widza, np. dodanie wstrząsu kamery przy uderzeniu
6. **Inne:**
- a. Opowiadana historia gra kluczową rolę, wciąga gracza
 - b. Oprawa graficzna powinna być na odpowiednim poziomie
 - c. Gracz musi być postawiony przed wyzwaniem

5. Charakterystyka informatycznych systemów mobilnych.

http://wazniak.mimuw.edu.pl/images/4/43/Systemy_mobilne_wyklad_1.pdf

http://wazniak.mimuw.edu.pl/index.php?title=Systemy_mobilne

Olbrzymi postęp technologiczny spowodował dostępność zaawansowanych i niedrogich urządzeń przenośnych i nawigacyjnych oraz systemów łączności bezprzewodowej.



- Rozwój technologiczny
- Miniaturyzacja
- Niski pobór prądu – długa praca na bateriach
- Nowoczesne łącza bezprzewodowe
- Spadek cen
- Powszechna dostępność

Nowe technologie, miniaturyzacja i coraz większa dostępność nowoczesnych urządzeń, mobilnych i bezprzewodowych pozwala na budowę systemów o nieznanych dotąd możliwościach.

Mały rozmiar tych urządzeń, ich niski pobór prądu, programowalność oraz odporność na zmienne warunki środowiska umożliwia szerokie ich zastosowanie w trudnych warunkach terenowych.

Bezprzewodowe interfejsy nie wymagają budowy złożonej i kosztownej infrastruktury, co znacznie przyspiesza wdrożenie i ewentualne zmiany w projektowanych systemach.



Zalety systemów mobilnych

- Dostęp do danych niezależnie od miejsca i czasu,
- Skrócony czas dostępu do danych (czas reakcji),
- Skalowalność – możliwość ciągłego rozwoju systemu w zależności od zadań,
- Efektywność wykorzystania zasobów – możliwość współdzielenia zasobów niezależnie od fizycznej lokalizacji użytkownika i zasobu,
- Łatwość realizacji nowych usług, w tym multimedialnych.

Stosowanie systemów mobilnych w firmie ma wiele zalet. Podstawową z nich jest nie podlegający dyskusji dostęp do danych niezależnie od miejsca i czasu. Związany jest z tym skrócony czas dostępu do danych (czas reakcji), co może mieć olbrzymie znaczenie dla administratorów systemów komputerowych czy produkcyjnych jak i podejmującej błyskawiczne decyzje, przebywającej poza firmą kadry kierowniczej. Systemy mobilne są w pełni skalowalne, czyli potrafią dynamicznie dostosowywać się do ciągłego wzrostu potrzeb i zadań, jakie przed nim stoją. Inną ważną cechą jest efektywność wykorzystania zasobów, czyli możliwość współdzielenia zasobów niezależnie od fizycznej lokalizacji użytkownika i zasobu. Łatwość realizacji nowych usług, w tym multimedialnych jest najważniejszą chyba, z punktu widzenia użytkownika końcowego, ich zaletą.



Cechy charakterystyczne systemu mobilnego

- Brak wspólnej pamięci
- Tylko wymiana wiadomości (message passing)
- Brak globalnego zegara
- Asynchronizm przetwarzania
- Asynchronizm komunikacji
- Przetwarzanie rozproszone

System mobilny, jak klasycznym system rozproszonym, złożonym z węzłów na których odbywa się przetwarzanie, połączonych siecią komunikacyjną – najczęściej bezprzewodową. Różni się tym od tradycyjnego systemu rozproszonego tym, że użytkownik zmienia swoje położenie, co powoduje ciągłą zmianę topologii sieci.

Podobnie jak w klasycznym systemie rozproszonym i tutaj występuje całkowity brak wspólnej pamięci, każdy węzeł dysponuje pamięcią lokalną. Komunikacja między węzłami i przekazywanie informacji odbywa się tylko na zasadzie wymiany wiadomości. W systemie mobilnym nie istnieje globalny zegar, podejmuje się jedynie próby synchronizacji zegarów kwarcowych zainstalowanych na komputerach mobilnych. Przetwarzanie rozproszone, jakie ma miejsce w takim systemie charakteryzuje również całkowity asynchronizm, podobnie komunikacja między węzłami odbywa się w sposób całkowicie asynchroniczny.



- Mały ekran- trudna prezentacja wyników przetwarzania
- Potrzeba innego podejścia do problemu
- Inne interfejsy z użytkownikiem (brak myszki, klawiatury)
- Inne warunki pracy (las, pole bitwy, bocznica kolejowa)

6. Definicja i modele przetwarzania dużych danych (ang. Big Data).

Big data – termin odnoszący się do dużych, zmiennych i różnorodnych zbiorów danych, których przetwarzanie i analiza jest trudna, ale jednocześnie wartościowa, ponieważ może prowadzić do zdobycia nowej wiedzy.

Pojęcie dużego zbioru danych jest względne i oznacza sytuację, gdy zbioru nie da się przetwarzać przy użyciu trywialnych, powszechnie dostępnych metod^[2]. W zależności od branży i stopnia złożoności algorytmu może to oznaczać rozmiar terabajtów lub petabajtów (np. analiza zderzeń cząstek elementarnych w fizyce wysokich energii^[3]), jednak w innych zastosowaniach będą to już megabajty bądź gigabajty (np. porównywanie billingów telefonicznych w telekomunikacji^[4]). *Big data* ma zastosowanie wszędzie tam, gdzie dużej ilości danych cyfrowych towarzyszy potrzeba zdobywania nowych informacji lub wiedzy. Szczególne znaczenie odgrywa wzrost dostępności Internetu oraz usług świadczonych drogą elektroniczną, które w naturalny sposób są przystosowane do wykorzystywania baz danych. Wykorzystanie do analiz dużych zbiorów danych oznacza jednocześnie, że nie trzeba ograniczać się do mniejszych

zbiorów określanych za pomocą różnych sposobów doboru próby, co eliminuje związane z tym błędy^[5].

W 2001 roku META Group opublikowała raport^[6], który opisuje *big data* w modelu 3V:

- duża ilość danych (ang. *volume*);
- duża prędkość przetwarzania danych (ang. *velocity*);
- duża różnorodność danych (ang. *variety*).

Model ten uzupełniony został o kolejne składowe – weryfikację posiadanych danych (ang. *veracity*) oraz wartość dla użytkownika (ang. *value*)^{[5][7]}.

Zastosowanie modelu w polskiej wersji 4W przedstawia się następująco^[8]:

- wykorzystanie – wykorzystaj najpierw wewnętrzne (własne) zasoby danych;
- wnioskowanie – umiejętnie stosuj techniki analityczne, użyj ekspertów;
- wzbogacanie – wzbogacaj własne dane o informacje z rynku, używaj słowników i baz referencyjnych;
- weryfikacja – koniecznie weryfikuj hipotezy i wnioski.

[\[https://pl.wikipedia.org/wiki/Big_data\]](https://pl.wikipedia.org/wiki/Big_data)

7. Definiowanie schematów dokumentów XML.

Różnice między DTD, a XML-Schema.

XML (ang. Extensible Markup Language, w wolnym tłumaczeniu Rozszerzalny Język Znaczników) – uniwersalny język znaczników przeznaczony do reprezentowania różnych danych w strukturalizowany sposób. Jest niezależny od platformy, co umożliwia łatwą wymianę dokumentów pomiędzy heterogenicznymi (różnymi) systemami i znacząco przyczyniło się do popularności tego języka w dobie Internetu. XML jest standardem rekomendowanym oraz spacyfikowanym przez organizację W3C.

XML stanowi podzbiór SGML-a. W założeniach ma pozwalać na wysyłanie, odbieranie i przetwarzanie danych rdzennego SGML w sferze www – ogólnie, zapewniać interoperacyjność między SGML a HTML. W przypadku wspólnie obsługiwanych funkcji, dane dokumenty SGML i XML są równoważne. W przypadku używania funkcji niedozwolonych w XML, możliwa jest transformacja dokumentu z zachowaniem danych.

DTD jako opis dokumentu XML: - pozwala stworzyć jedynie bardzo ogólny schemat dokumentu - w standardzie DTD dostępne mamy jedynie rozróżnienie na elementy proste i złożone - wewnątrz elementów prostych występuje tylko jeden typ danych, wewnątrz złożonych nie możemy dokładnie określić liczby wystąpień elementu

XML Schema – zastępuje DTD uzupełniając go o wiele nowych opcji: - możemy dokładnie określać typy i wzorce zawartości elementów a także atrybutów - wskazujemy też ilość i sekwencje występowania poszczególnych elementów

W praktyce ze względu na małe możliwości obecnie DTD jest wypierane przez nowocześniejsze [XML Schema](#), które posiada znacznie większe możliwości oraz nie wymaga stosowania dodatkowej, nie-XML-owej składni.

Zasadnicza różnica między DTD a XML Schema polega na tym, że XML Schema wykorzystuje składnię opartą na XML, podczas gdy DTD mają unikalną składnię z DTD SGML.

DTD

```
<!ELEMENT pracownicy (pracownik+)>
<!ELEMENT pracownik (daneOsobowe, dział?)>

<!ELEMENT daneOsobowe (imie, drugieImie?, nazwisko)>
<!ELEMENT imie (#PCDATA)>
<!ELEMENT drugieImie (#PCDATA)>
<!ELEMENT nazwisko (#PCDATA)>

<!ELEMENT dział (#PCDATA)>
```

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE pracownicy SYSTEM "pracownicy.dtd">
<pracownicy>
  <pracownik>
    <daneOsobowe>
      <imie>Izabela</imie>
      <nazwisko>Nowakowska-Jasnorzębska</nazwisko>
    </daneOsobowe>
  </pracownik>
  <pracownik>
    <daneOsobowe>
      <imie>Zygfried</imie>
      <drugieImie>Zenobiusz</drugieImie>
      <nazwisko>Wawrzyniak</nazwisko>
    </daneOsobowe>
    <dział>Public Relations</dział>
  </pracownik>
</pracownicy>
```

Różnice między XML Schema Definition (XSD) a Document Type Definition (DTD) obejmują:

- XML Schema są zapisane w XML, a DTD pochodzą ze składni SGML.
- XML Schema definiują typy danych dla elementów i atrybutów, podczas gdy DTD nie obsługuje typów danych.
- XML Schema umożliwiają obsługę przestrzeni nazw, podczas gdy DTD tego nie robi.
- XML Schema definiują liczbę i kolejność elementów potomnych, podczas gdy DTD nie.
- XML Schema można manipulować z XML DOM, ale nie jest to możliwe w przypadku DTD.
- za pomocą XML Schema użytkownik nie musi uczyć się nowego języka
- Schemat XML zapewnia bezpieczną komunikację danych, np. nadawca może opisywać dane w sposób zrozumiały dla odbiorcy, ale w przypadku DTD dane mogą być źle zrozumiane przez odbiorcę.
- XML Schema są rozszerzalne, a DTD nie jest.

Z drugiej strony:

- DTD umożliwia zdefiniowanie nowych wartości ENTITY do użycia w pliku XML.
- DTD pozwala rozszerzyć go lokalnie do pojedynczego pliku XML.

No.	DTD	XSD
1)	DTD stands for Document Type Definition .	XSD stands for XML Schema Definition.
2)	DTDs are derived from SGML syntax.	XSDs are written in XML.
3)	DTD doesn't support datatypes .	XSD supports datatypes for elements and attributes.
4)	DTD doesn't support namespace .	XSD supports namespace .
5)	DTD doesn't define order for child elements.	XSD defines order for child elements.
6)	DTD is not extensible .	XSD is extensible .
7)	DTD is not simple to learn .	XSD is simple to learn because you don't need to learn new language.
8)	DTD provides less control on XML structure.	XSD provides more control on XML structure.

<https://www.sitepoint.com/xml-dtds-xml-schema/>

8. Harmonogramowanie projektów informatycznych.

[z wykładu 4 ZPI Trawińskiego]

Harmonogramowanie to wybór i zastosowanie najbardziej odpowiednich technik dla stworzenia programu lub sekwencji działań nakierowanych na dotrzymanie kluczowych terminów i celów projektu. Obejmuje ono również rozpoznanie i uwzględnienie w programie:

- etapów (faz), dat, punktów kontrolnych (kamieni milowych),
- zapotrzebowania i dostępności zasobów,
- sekwencji zadań i zależności pomiędzy zadaniami,
- limitów czasowych,
- ograniczeń wewnętrznych i zewnętrznych.

Techniki harmonogramowania:

- wykres Gantt'a lub wykresy paskowe,
- diagramy sieciowe (zadanie na wierzchołku/łuku),
- metoda ścieżki krytycznej,
- harmonogramowanie zasobów,
- harmonogramowanie punktów kontrolnych (kamieni milowych).

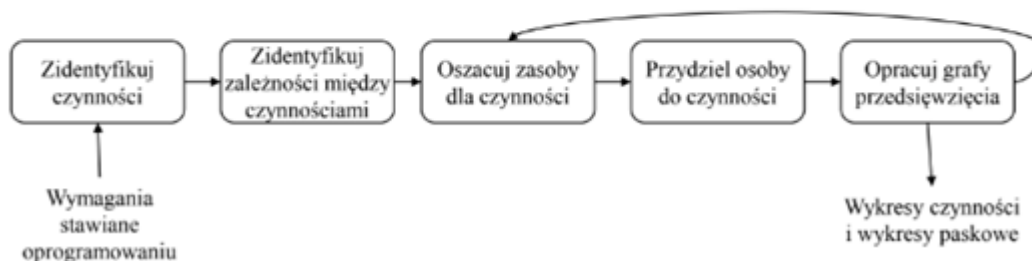
Tworzenie harmonogramu przedsięwzięcia:

- podziel przedsięwzięcie na zadania i oszacuj czas i zasoby konieczne do wykonania każdego z nich,
- zorganizuj zadania równolegle aby najlepiej wykorzystać ludzi i zasoby,
- minimalizuj zależności pomiędzy zadaniami, aby najlepiej wykorzystać czas przypadający na każde z nich,
- wszystko zależy od umiejętności i intuicji zarządzającego projektem.



Technika
Łódzka

Proces tworzenia harmonogramu przedsięwzięcia



9. Heurystyki użyteczności Nielsena.

W 1990 wraz z Ralfem Molichem Nielsen przedstawił 10 heurystyk, czyli zasad interakcji człowiek-maszyna

1. Pokazuj status systemu
2. Zachowaj zgodność pomiędzy systemem a rzeczywistością
3. Daj użytkownikowi pełną kontrolę
4. Trzymaj się standardów i zachowaj spójność
5. Zapobiegaj błędom
6. Pozwalaj wybierać zamiast zmuszać do pamiętania
7. Zapewnij elastyczność i efektywność
8. Dbaj o estetykę i umiar
9. Zapewnij skuteczną obsługę błędów
10. Zadbaj o pomoc i dokumentację

1. Pokazuj status systemu

Użytkownik czuje się lepiej, jeżeli wie co się dzieje i gdzie w danym momencie się znajduje. Dzięki temu w trakcie użytkowania witryny czuje się pewniej. Informacja o tym w jakiej części serwisu znajduje się użytkownik albo na jakim etapie składania zamówienia jest więc co najmniej wskazana. Dobrym sposobem na lokalizowanie miejsc w których powinien być pokazywany status systemu jest analizowanie witryny pod kątem procesów realizowanych przez użytkownika (proces składania zamówienia, proces wyszukiwania produktu, proces dopisania do newslettera). Jeżeli mamy jakiś proces to możemy też określić na jakim etapie danego procesu jesteśmy (a tym samym powinniśmy pokazać status). Innym dobrym przykładem zastosowania tej zasady są tak zwane „okruszki” (ang. breadcrumbs) stale pokazujące użytkownikowi w jakiej części serwisu obecnie się znajduje. Dzięki „okruszkom” użytkownik może łatwo „cofnąć się” bez używania przycisku „wstecz” przeglądarki.

2. Zachowaj zgodność pomiędzy systemem a rzeczywistością

Inaczej: **Używaj języka, który jest zrozumiały poza twoim systemem.** Jest to jedna z bardziej enigmatycznych zasad. W wielkim skrócie—strona ma być zrozumiała dla każdego użytkownika internetu (a nie tylko dla absolwentów kierunków technicznych). Najczęściej w przypadku tej heurystyki mówi się po prostu o używanym języku. Przykładowo jeżeli sklep internetowy sprzedaje odbiorniki cyfrowej telewizji naziemnej to ich opisy mają zostać przygotowane tak, żeby osoba niespecjalnie interesująca się tym zagadnieniem potrafiła samodzielnie wybrać najlepszy dla siebie produkt.

3. Daj użytkownikowi pełną kontrolę

Każdy użytkownik powinien mieć kontrolę nad operacjami, które wykonuje w obrębie naszego serwisu. Przykładowo jeżeli dodał on coś do koszyka, to powinien mieć on możliwość łatwego (intuicyjnego) usunięcia tego towaru z koszyka. Podobnie jeżeli ktoś szuka produktu i „schodzi” do coraz bardziej szczegółowych podkategorii produktowych to powinien mieć możliwość powrotu do bardziej ogólnej kategorii bez konieczności klikania przycisku „wstecz” w swojej przeglądarce.

4. Trzymaj się standardów i zachowaj spójność

Są pewne standardy, których nieprzestrzeganie jest niczym strzelanie sobie w kolano. Przykładowo, logotyp musi być umieszczony z lewej strony na samej górze oraz być podlinkowany do strony głównej, wewnętrzną wyszukiwarkę umieszcza się raczej w prawym górnym rogu, a link do regulaminu znajdziemy prawie zawsze na dole strony. Pisząc o spójności Jakob Nielsen ma na myśli konsekwencję w zastosowaniu rozwiązań w obrębie danej strony. Najlepszym tego przykładem może być zastosowanie identycznego formatowania dla wszystkich linków, przycisków z nawołaniem do podjęcia akcji (ang. call to action) oraz nagłówków. Dzięki przestrzeganiu tej zasady, użytkownicy nie będą zmuszeni do długiej nauki obsługi naszej witryny. Wystarczy, że raz zorientują się, jakie elementy są możliwe do kliknięcia i dzięki temu, że inne podstrony są spójne to będzie im łatwiej poruszać się po całej witrynie.

5. Zapobiegaj błędom

Możemy spotkać się z najróżniejszymi interpretacjami tej zasady. W mojej ocenie Jakob Nielsen miał na myśli nie doprowadzanie do sytuacji, w których użytkownik będzie miał wrażenie, że coś zadziało nie tak. Przykładem może być pokazywanie w sklepie internetowym produktów, których i tak nie można zamówić (nie ma ich w magazynie, nie jest obecnie możliwe zamówienie ich u producenta, itd.). Nie możemy doprowadzać do sytuacji, że użytkownik dokładnie zapoznał się z opisem produktu, poświęcił 2 godziny na porównanie go z innymi alternatywami po czym okazuje się, że dany towar nie jest możliwy do zamówienia. Jeszcze innym bardzo dobrym przykładem może być działanie filtrowania w obrębie strony. Wyobraźmy sobie, że mamy sklep internetowy sprzedający odkurzacze i w opcjach filtrowania możemy wybrać kolor. Wśród kolorów możliwych do wybrania jest pomarańczowy. Wybieramy ten kolor po czym okazuje się, że otrzymujemy pusty wynik wyszukiwania (sklep nie ma żadnego odkurzacza w kolorze pomarańczowym)—po co więc dawać możliwość wyboru opcji, która zaprowadza nas w ślepią uliczkę?

6. Pozwalaj wybierać zamiast zmuszać do pamiętania

Po prostu nie oczekujemy, że użytkownicy naszych witryn będą pamiętali o wyborach dokonywanych we wcześniejszych etapach interakcji z naszą stroną. Najlepiej, jeżeli strona jest skonstruowana w taki sposób, że użytkownik nie musi w trakcie interakcji z nią przypominać sobie różnych faktów. Jeżeli interakcje użytkownika ze stroną przerywane są koniecznością przypomnienia sobie czegoś, albo powrotu na wcześniej oglądaną podstronę, po jakąś niezbędną informację to znaczy, że z naszą stroną jest coś nie tak. Wróćmy do przykładu sklepu sprzedającego odkurzacze. Jeżeli w karcie produktu podamy wagę produktu i następnie w procesie składania zamówienia uzależnimy koszt dostawy od wagi produktu, przy czym użytkownicy będą musieli na tym etapie sami wpisać wagę produktu, to jest to oczywisty błąd. Obecnie bardzo rzadko spotykamy się z tego rodzaju błędami (między innymi dzięki rozwijającym listom), jednak od czasu do czasu możemy zobaczyć np. formularze, których opisy poszczególnych pól umieszczone są właśnie wewnątrz pól. Po rozpoczęciu wpisywania tekstu, opis znika a część użytkowników może zapomnieć czego dotyczyło dane pole.

7. Zapewnij elastyczność i efektywność

Witryna internetowa nie może idealnie spełniać potrzeb wszystkich użytkowników jednak w zakresie niektórych funkcjonalności powinna być elastyczna. Przykładem może być

zaawansowana wyszukiwarka produktów w przypadku sklepu internetowego. Pamiętajmy też, że różne osoby w różny sposób wyszukują informacji na stronie internetowej (niektórzy korzystają z menu, inni z wyszukiwarki a jeszcze inni szukają graficznych pomocy)—w obrębie witryny nie może więc zabraknąć możliwości poruszania się po stronie przy użyciu którejkolwiek z opisanych funkcjonalności. Ważne jest także aby zapewnić możliwość zmiany wielkości czcionki (zaskakujące jest jak wiele osób korzysta z tej funkcjonalności). Mówiąc o efektywności koniecznie trzeba wspomnieć o czasie wczytywania strony, który ma duży wpływ na współczynnik konwersji strony internetowej.

8. Dbaj o estetykę i umiar

Ludzie chętniej korzystają z rzeczy, które im się podobają. Właśnie dlatego tak ważny jest design. Strona internetowa nie musi być dziełem sztuki, ale powinna podobać się użytkownikom. Pamiętajmy przede wszystkim o jednej z naczelnych zasad dobrego designu—o kontraście. Dzięki niemu nawet przy dużej liczbie elementów na stronie możemy w łatwy i prosty sposób wskazać użytkownikowi, które rzeczy są najważniejsze i godne uwagi.

9. Zapewnij skuteczną obsługę błędów

Tak naprawdę użytkownik nie powinien mieć do czynienia z błędami—w końcu jedna z wcześniejszych heurystyk mówi, że mamy ich unikać. Jeżeli jednak błędy się pojawiają to, po pierwsze nie atakujemy użytkowników technicznym słownictwem, po drugiej nie piszmy, że błąd jest wynikiem nieodpowiednich działań użytkownika i po trzecie napiszmy, co użytkownik ma zrobić dalej. Strona błędu 404 jest idealnym elementem, na którym powinniśmy zastosować te zasady.

10. Zadbaj o pomoc i dokumentację

Ostatnia heurystyka jasno pokazuje, że nawet jeżeli spełnimy wszystkie wcześniejsze zasady to i tak część użytkowników będzie potrzebowała pomocy w obsłudze strony. Najczęściej na samym dole strony znajdujemy linki do odpowiednich działów pomocy—jest to stosunkowo dobre rozwiązanie umożliwiające w prosty i szybki sposób odnaleźć odpowiedzi na nurtujące nas pytania. Pamiętajmy tylko o tym, że nawet jeżeli nasza strona jest idealna to i tak trzeba założyć, że niektórzy użytkownicy mogą mieć problemy w trakcie korzystania z niej.

10. Indeksowanie i wyszukiwanie informacji multimedialnych.

<https://docplayer.pl/11411460-Sposoby-wyszukiwania-multimedialnych-zasobow-w-internecie.html>

Multimedia są środkami komunikacji wykorzystującymi różne formy informacji oraz różne formy ich przekazu w celu dostarczania odbiorcom informacji lub rozrywki. Do najważniejszych form przekazu należą:

- tekst;
- dźwięk, w tym muzyka;
- obrazy, w tym grafika i zdjęcia oraz film i animacja.

Multimedia to połączenie przynajmniej dwóch z wymienionych form przekazu, a więc może to być np. zdjęcie połączone z tekstem (metadanymi) czy wideo (połączenie dźwięku i obrazu, a jeśli występują napisy to i tekstu). Do multimediiów możemy również zaliczyć prezentacje Power Point, w których można wykorzystać niemal wszystkie z wymienionych form przekazu. Z tego samego powodu multimediami będą też aplikacje interaktywne, np. wszelkie techniki szkolenia stosowane w e-learningu.

Przykłady multimediiów:

- zdjęcia
- obrazy
- animacje
- filmy video
- audycje radiowe
- audycje telewizyjne
- podcasty
- interaktywne kursy, tutoriale
- mapy interaktywne
- prezentacje

Sposoby indeksowania multimediiów do celów wyszukiwawczych

Każdy, kto miał do czynienia z przygotowywaniem jakichkolwiek materiałów do uczynienia ich „widzialnymi” przez system wyszukiwawczy (np. katalogi biblioteczne, bazy danych) zdaje sobie sprawę, ile trudu trzeba włożyć w doskonałe opisanie dokumentu. I dokładnie tak samo trzeba postępować z multimediami, żebyśmy mogli łatwo do nich dotrzeć. Jednak o ile z dokumentami tekstowymi nie jest to trudne, bo wystarczy „wyciągnąć” z tekstu odpowiednie informacje, a resztę zrobi za nas maszyna, która w najlepszym przypadku dokona wyszukiwania pełnotekstowego, to w przypadku multimediiów nie jest to takie proste. Dla programu indeksującego zawartość i potem wyszukującego - wg pytania zadanego przez użytkownika – jest to problemem. Użytkownik – przynajmniej na razie – zadaje pytanie tekstowe, które nie przekłada się na zawartość np. filmu. Tak więc, systemy wyszukiwawcze bez trudu radzą sobie z odnajdywaniem danych tekstowych, nawet tych nieskatalogowanych, ale nie potrafią „zagłębić” się w zawartość multimediiów tak, żeby odpowiedzieć na nasze pytanie. Chyba, że moglibyśmy nasz problem zamiast opisywać – narysować, namalować bądź zanucić? Póki to nie będzie możliwe, kluczowym problemem jest właściwe i dokładne opisywanie materiałów multimedialnych zamieszczanych w sieci i wdrażanie nowoczesnych rozwiązań

automatycznego wnikania w ich zawartość, tak jak przy pomocy techniki OCR można automatycznie rozpoznawać znaki i indeksować dokumenty tekstowe.

Jakie są obecnie powszechne metody indeksowania i wyszukiwania multimedialnych? Robot wyszukiwarki ogólnej (np. Google) „odsiewa” strony, na których znajdują się obrazki/wideo/audio na podstawie kodu źródłowego. Następnie analizuje tekst znajdujący się na tej stronie i indeksuje na ogół następujące elementy: nazwę pliku (np. obrazka), podpis, tekst towarzyszący plikowi, metadane i inne parametry (np. rozmiar pliku, kolor, format). Jeśli zatem hasło wpisane przez użytkownika nie występuje w w/w elementach, standardowa wyszukiwarka nie odnajdzie zasobu. Mamy do czynienia również z innymi sytuacjami – szukane wyrażenie znajduje się w tekście na stronie, w pobliżu np. zdjęcia zupełnie nie związanego z wyszukiwaniami i takie zdjęcie pojawi się na liście rezultatów. Trochę lepiej sytuacja wygląda w wyszukiwarkach specjalistycznych i bazach danych, gdzie każdy zasób multimedialny jest katalogowany przez człowieka (metadane są różnicowane, zależności od serwisu). Poza tym mamy gwarancję, że znajdziemy wyłącznie interesujące np. obrazy lub wideo, a nie stronę WWW gdzie przypadkowo znalazło się zdjęcie i to nie związane z naszymi poszukiwaniami. Inną metodą podniesienia jakości indeksowania i wyszukiwania multimedialnych jest włączanie w ten proces użytkowników sieci. Dotyczy to tzw. serwisów społecznościowych, których zawartość jest tworzona i doskonalona przez samych internautów. W takich serwisach jak np. YouTube użytkownicy dodają nagrania wideo i dodatkowe informacje: tytuł, opis i słowa kluczowe (tzw. tagi), które podlegają wyszukiwaniu. Jakość tagów może pozostawiać wiele do życzenia, zwłaszcza dla czułych pod tym względem bibliotekarzy. Musimy jednak docenić fakt, że katalogowanie zasobów przez „zwykłych” ludzi (folksonomia) będzie się rozwijać i niekiedy dawać lepsze rezultaty, bo katalogerami są ci, którzy jednocześnie wyszukują potrzebne informacje, a więc mówią tym samym językiem.

Powszechnie stosowane obecnie wyszukiwarki/serwisy są zatem nadal nastawione na wyszukiwanie tekstowe. Jednak ogromny wzrost treści multimedialnych i zapotrzebowanie na głębsze „wnikanie” w te treści niż tylko przeszukiwanie opisów, tytułów i innych metadanych tekstowych, doprowadziły do poszukiwania nowych rozwiązań, z których większość istnieje w fazie eksperymentalnej albo dostępnej komercyjnie. Poza obecnie stosowanymi technikami indeksowania, kolejnym etapem będzie użycie nie tylko informacji tekstowych towarzyszących multimedialnym, ale również:

- analiza ścieżki dźwiękowej (wykrywanie i rozpoznawanie mowy, przekształcenie mowy na tekst, rozpoznawanie dźwięków itd.),
- analiza obrazów (kolor, kształt, tekstura, wykrywanie i rozpoznawanie twarzy),
- analiza wideo (w/w oraz rozpoznawanie klatki filmu, wytwórni, wykrywanie i rozpoznawanie twarzy, rozpoznawanie tekstu w filmie itd.).

Niektóre pliki multimedialne same w sobie zawierają tekst, który można rozpoznawać przy pomocy OCR, np. audycje telewizyjne, które zawierają tekst cyfrowy jako napisy na ekranie i tytuły scen, napisy z wideo. Rozwijają się również systemy rozpoznawania mowy i przekształcania mowy na tekst, które znajdują zastosowanie w automatycznym indeksowaniu wszelkiego rodzaju transmisji dźwiękowych. Ta technologia pozwala na wyszukiwanie „słowa

mówionego” w treści np. audycji radiowej czy fragmentów nagrań wideo. Podczas konwersji wypowiedzianych słów na tekst (nie zawsze „słowo w słowo”) z użyciem algorytmów rozpoznawania mowy, powstaje indeks dla mowy występującej w cyfrowych plikach audio i wideo, który służy do rozpoznawania kolejnych plików dźwiękowych oraz wyszukiwania całego zasobu. Każde słowo lub głoska opatrzona zostaje kodem, umożliwiającym identyfikację odpowiednich segmentów cyfrowych danych audiowizualnych. W ten sposób wszelkie zapisy „mówione” mogą być „słyszane” przez wyszukiwarki. Nagrania wideo są indeksowane głównie na podstawie analizy ścieżki dźwiękowej, ale w zaawansowanych systemach również analizy obrazu i tworzenia metadanych (twórcy, programy, miejsce, temat). Wszystkie te dane tworzą indeks wyszukiwarki/bazy danych.

Techniki rozpoznawania obrazów koncentrują się głównie wokół kształtu, kolorystyki i tekstury. Zaawansowane techniki pozwalają użytkownikom „narysować” kształt poszukiwanego obiektu albo wybrać przykładowy obraz i podobne. Metody rozpoznawania twarzy na podstawie obrazów działają w oparciu o wygląd (charakterystyczne cechy twarzy) lub model twarzy np. kontury oczu, nosa, policzków. Kombinacja tak określonego kształtu twarzy wraz z teksturą obrazu pozwala na obliczenie podobieństwa między dwoma twarzami.

Kolejnym rozwiązaniem są narzędzia wyszukiwawcze, które filtrują rezultaty dla plików muzycznych. Dzięki dostawcom cyfrowych nagrań muzycznych ścieżka dźwiękowa ma już wbudowane takie metadane jak nazwisko autora, nazwę ścieżki, nazwę albumu i gatunek, a niektóre generują również słowa piosenki czy nazwisko członka zespołu. Jednak firmy nagrywające formatują swoje dane w zróżnicowany sposób, co stanowi utrudnienie dla wyszukiwarek. Wyszukiwarki i przemysł muzyczny przewidują, że w przyszłości użytkownik będzie mógł wpisać słowa piosenki lub zanucić kilka jej taktów i otrzymać na ekranie plik muzyczny lub wideo, nawet jeśli nie zna tytułu czy autora.

Sposoby wyszukiwania multimedialnych zasobów w internecie

W czasie, gdy w sieci zaczęło pojawiać się coraz więcej zasobów multimedialnych i ich „odsianie” spośród innych informacji stało się zbyt czasochłonne i kłopotliwe dla użytkowników, zaczęły powstawać wyspecjalizowane narzędzia do wyszukiwania multimedialnych zasobów. Można do nich zaliczyć:

- specjalne opcje standardowych wyszukiwarek;
- specjalistyczne wyszukiwarki wszystkich multimedialnych zasobów w sieci albo wybranego typu, np. zdjęć;
- kolekcje wybranego typu zasobów multimedialnych tworzone przez wyspecjalizowane firmy, biblioteki czy wspólnie przez internautów (tzw. serwisy społecznościowe);
- repozytoria, katalogi tematyczne (subject gateways), w których spotyka się kategorie „multimedia”.

Korzystając z tych wszystkich narzędzi trzeba pamiętać o rzeczy podstawowej – właściwym formułowaniu pytania. Pamiętając o opisanych wyżej niedoskonałościach wyszukiwarek multimedialnych i sposobach indeksowania zawartości, zadawajmy pytania myśląc podobnie jak maszyna czy człowiek, który zindeksował obrazy, filmy i nagrania. A przede wszystkim korzystajmy, gdzie to tylko możliwe, z zaawansowanych funkcji wyszukiwania, operatorów logicznych i innych przydatnych metod.

1) Standardowe wyszukiwarki

Wiele wyszukiwarek ogólnych, np. Google, Yahoo!, polskie Interneta i Goooru, umożliwia wyszukiwanie zasobów multimedialnych poprzez odrębną zakładkę w formularzu do wyszukiwania, np. Images, Video, Audio, Grafika. Niektóre z nich uruchomiły odrębny interfejs, np. Google video. W każdym przypadku mamy do dyspozycji wyszukiwanie proste i zaawansowane. W tym drugim można ograniczać wyszukiwanie wg różnych kategorii, np. formatu pliku, kolorystyki, rozmiaru, kategorii tematycznej, czasu trwania nagrania. Zróżnicowane są też bazy danych wyszukiwarek – niektóre przeszukują własną kolekcję zdjęć/wideo/audio czyli te zasoby, które odnalazły w sieci ich roboty; inne do własnych kolekcji dodają bazy danych wyszukiwarek specjalistycznych, np. nagrań wideo YouTube. W drugim przypadku w prezentowanych wynikach, na miniaturce wideo pojawia się logo właściciela kolekcji (zob. Google video). Wyszukując zasoby multimedialne przy pomocy ogólnych wyszukiwarek musimy jednak pamiętać o wspomnianych wyżej ograniczeniach. Narzędzia te nie są jeszcze na tyle doskonałe, żeby wnikać „do wnętrza” multimedialnych zasobów, do ich właściwej treści. W wynikach wyszukiwania otrzymamy zatem najczęściej źródła, w których występuje plik nazwany tak, jak poszukiwane przez nas wyrażenie albo po prostu tekst czy metadane z tym wyrażeniem, a obok niego zdjęcie/wideo/audio, nie zawsze na temat.

2) Specjalistyczne wyszukiwarki multimedialnych

Innym sposobem wyszukiwania multimedialnych zasobów jest skorzystanie ze specjalnych wyszukiwarek ograniczających się do przeszukiwania wyłącznie tego typu zasobów. Mogą to być zarówno wyszukiwarki wszystkich zasobów czyli obrazów, wideo i audio łącznie lub wyspecjalizowane wyszukiwarki, np. tylko zdjęć. Podobnie jak w przypadku wyszukiwarek ogólnych, narzędzia te bazują na własnych kolekcjach albo przeszukują całą sieć pod kątem np. audycji radiowych. Niektóre z nich wykorzystują zaawansowane technologie rozpoznawania mowy czy obrazów. Stąd wyniki wyszukiwania są dużo bardziej precyzyjne niż w wyszukiwarkach ogólnych.

11. Inteligencja obliczeniowa - metody i obszary zastosowań.

Inteligencja Obliczeniowa (ang. Computational Intelligence, CI)

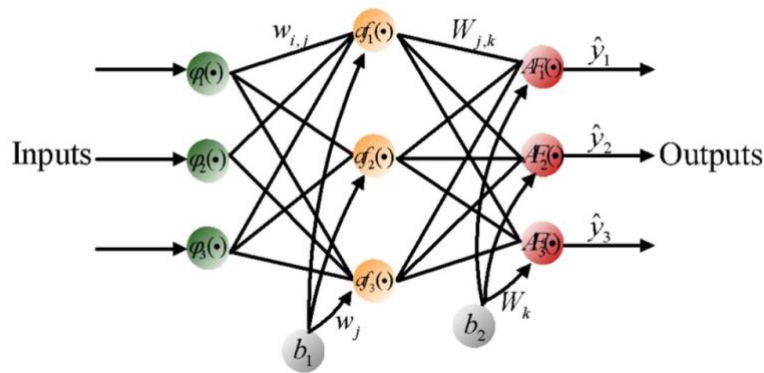
- Zbiór inspirowanych biologicznie metod uczenia się
- Budowa modeli procesów rzeczywistych
- Wspólna cecha - uczenie z niedokładnych, niepewnych lub częściowych danych
- Stosowane tam, gdzie klasyczne metody (ręcznego) modelowania nie sprawdzają się



Sieci neuronowe

<http://kognitywistyka.uwb.edu.pl/component/k2/item/406-sieci-neuronowe>

Sieć neuronowa to połączenie elementów zwanych sztucznymi neuronami, które tworzą co najmniej trzy warstwy: wejściową, ukrytą i wyjściową, przy czym warstw ukrytych może być wiele. Neurony sieci przetwarzają informacje dzięki temu, że ich połączeniom nadaje się parametry, zwane wagami, które modyfikuje się podczas działania sieci. Modyfikowanie wag nazywane jest „uczeniem się” sieci. W procesie uczenia się sieci tkwi istota i rzeczywista wartość sieci neuronowych.



Logika rozmyta

https://pl.wikipedia.org/wiki/Logika_rozmyta

Logika rozmyta – jedna z logik wielowartościowych, stanowi uogólnienie klasycznej dwuwartościowej logiki. Została zaproponowana przez Lotfi Zadeha, jest ściśle powiązana z jego teorią zbiorów rozmytych. W logice rozmytej między stanem 0 (fałsz) a stanem 1 (prawda) rozciąga się szereg wartości pośrednich, które określają stopień przynależności elementu do zbioru[1].

Logika rozmyta okazała się bardzo przydatna w zastosowaniach inżynierskich, gdzie klasyczna logika klasyfikująca jedynie według kryterium prawda/fałsz nie potrafi skutecznie poradzić sobie z wieloma niejednoznacznościami i sprzecznościami. Znajduje wiele zastosowań, między innymi w elektronicznych systemach sterowania (maszynami, pojazdami i automatami), zadaniach eksploracji danych czy też w budowie systemów ekspertowych.

Metody logiki rozmytej wraz z algorytmami ewolucyjnymi i sieciami neuronowymi stanowią nowoczesne narzędzia do budowy inteligentnych systemów mających zdolności uogólniania wiedzy.

Zbiór klasyczny – opiera się na dwóch wartościach logicznych: prawda i fałsz (1 lub 0). Pomędzy tymi wartościami istnieje zawsze jasno określona granica – coś może przyjmować tylko wartość 1 lub tylko wartość 0.



Algorytmy ewolucyjne

<http://edu.pjwstk.edu.pl/wyklady/nai/scb/wyklad10/w10.htm>

Algorytmy ewolucyjne to nazwa ogólna, obejmująca takie metody szczegółowe, jak np.:

- algorytmy genetyczne,
- programowanie genetyczne,
- strategie ewolucyjne.

Ich cechą wspólną jest wykorzystanie schematu działania wzorowanego na teorii doboru naturalnego i ewolucji. Jest to więc kolejna - po sieciach neuronowych - technika inspirowana analogiami biologicznymi. Główne pole zastosowań algorytmów (metod) ewolucyjnych to problemy optymalizacji - jak wiemy, wiele problemów praktycznych można przedstawić w języku problemów optymalizacyjnych; większość z nich można w związku z tym próbować rozwiązywać ewolucyjnie.

Zasadę działania algorytmów ewolucyjnych można nieformalnie streścić następująco: zamiast szukać na oślep (lub konstruować z kawałków, jak w przypadku strategii zachłannej) właściwego rozwiązania problemu, postaramy się je "wychodować". Niech różne rozwiązania konkurują ze sobą, krzyżują się, rozmnażają, naśladując mechanizmy odpowiadające za coraz lepsze dostosowanie organizmów żywych do środowiska. Skoro ewolucja naturalna jest tak skuteczna w konstruowaniu złożonych, dobrze dostosowanych organizmów, to możemy liczyć na to, że ewolucja symulowana przyniesie nam dobre (zbliżone do optymalnych) rozwiązania problemu. Musimy tylko zadbać o to, by nasze symulowane "środowisko" promowało te osobniki (tzn. rozwiązania problemu), które lepiej spełniają nasze kryteria optymalizacyjne.

Drążenie danych (ang. data mining)

https://pl.wikipedia.org/wiki/Eksploracja_danych

Eksploracja danych (spotyka się również określenie drążenie danych, pozyskiwanie wiedzy, wydobywanie danych, ekstrakcja danych) (ang. data mining) - jeden z etapów procesu odkrywania wiedzy z baz danych (ang. Knowledge Discovery in Databases, KDD). Idea eksploracji danych polega na wykorzystaniu szybkości komputera do znajdowania ukrytych dla człowieka (właśnie z uwagi na ograniczone możliwości czasowe) prawidłowości w danych zgromadzonych w hurtowniach danych.

Zastosowania:

<http://nauka.metodolog.pl/obszary-zastosowania-data-mining/>

- Sprzedaż detaliczna

Za pomocą data mining identyfikuje się zachowania klientów (co i ile kupują), odkrywa trendy i wzorce zakupów klienta, poprawia jakość obsługi klienta, osiąga się lepszą satysfakcję klienta, stosuje się lepszą politykę dystrybucji i transportu oraz zmniejsza się koszty przedsiębiorstwa. Techniki eksploracji danych w handlu detalicznym mają zastosowanie w:

- Segmentacji klientów: identyfikują grupę klientów i dopasowują każdego klienta do odpowiedniej grupy.
- Ustalaniu zachowań klientów: identyfikują wzorce zakupów klienta i określają , jakie następne produkty klient prawdopodobnie kupi.
- Utrzymaniu klienta: identyfikuje zwyczaje „zakupowe” klienta i dostosowuje ofertę produktów , cen i promocje.
- Analizie kampanii sprzedażowych: przewiduje skuteczność sprzedaży opierając się na pewnych czynnikach, takich jak rabaty czy reklamy.

- Bankowość

Istnieje wiele obszarów, w których data mining może być stosowane w sektorze finansowym jak np. rentowność klientów, analiza kredytowa, prawdopodobieństwo niewypłacalności, marketing, oszustwa transakcyjne, ranking inwestycji, optymalizacji portfela, zarządzanie gotówką, prognozowanie i cross selling. Główne przykłady zastosowań technik eksploracji danych w bankowości to:

- Scoring kredytowy: metoda oceny wiarygodności podmiotu ubiegającego się o kredyt bankowy.
- Segmentacja klientów: ustanowienie grup klientów i dodanie każdego nowego klienta do odpowiedniej grupy.
- Utrzymaniu klienta: identyfikuje zwyczaje „zakupowe” klienta i dostosowuje ofertę produktów (w tym wypadku kont, kredytów itp.)
- Przewidywanie rentowności klientów: rozpoznawanie wzorców, na podstawie różnych czynników, takich jak np. produkty stosowane przez klienta, aby przewidzieć jego rentowność.

- **Ubezpieczenia**

Firmy ubezpieczeniowe wykorzystują data mining do pozyskiwania nowych klientów, zachowania dotychczasowych klientów i wykonywaniem różnych klasyfikacji. Główne zastosowania data mining w ubezpieczeniach są następujące:

- Identyfikacja czynników ryzyka: czyli analiza czynników , takich jak historia reklamacji lub wzorce zachowań, które mogą mieć silniejszy lub słabszy wpływ na poziom ryzyka ubezpieczonego.
- Wykrywanie oszustw: ustalenie wzorców oszustwa i przeanalizowanie czynników, które wskazują na duże prawdopodobieństwo oszustwa roszczenia.
- Utrzymywanie i segmentacja klientów: ustalenie grup klientów i dodanie każdego nowego klienta do odpowiedniej grupy oraz określenie rabatów i pakietów, które mogłyby zwiększyć lojalność klienta.

Uczenie maszynowe

https://pl.wikipedia.org/wiki/Uczenie_maszynowe

Uczenie maszynowe, samouczenie się maszyn albo systemy uczące się (ang. machine learning) – dziedzina wchodząca w skład nauk zajmujących się problematyką SI (patrz sztuczna inteligencja). Jest to nauka interdyscyplinarna ze szczególnym uwzględnieniem takich dziedzin jak informatyka, robotyka i statystyka. Głównym celem jest praktyczne zastosowanie dokonań w dziedzinie sztucznej inteligencji do stworzenia automatycznego systemu potrafiącego doskonalić się przy pomocy zgromadzonego doświadczenia (czyli danych) i nabywania na tej podstawie nowej wiedzy.

Uczenie maszynowe jest konsekwencją rozwoju idei sztucznej inteligencji i metod jej wdrażania praktycznego. Dotyczy rozwoju oprogramowania stosowanego zwłaszcza w innowacyjnych technologiach i przemyśle. Odpowiednie algorytmy mają pozwolić oprogramowaniu na zautomatyzowanie procesu pozyskiwania i analizy danych do ulepszania i rozwoju własnego systemu.

Uczenie się może być rozpatrywane jako konkretyzacja algorytmu czyli dobór parametrów, nazywanych wiedzą lub umiejętnością. Służy do tego wiele typów metod pozyskiwania wiedzy oraz sposobów reprezentowania wiedzy.

Ma to zapewnić zwiększenie:

- efektywności
- wydajności
- bezawaryjności
- redukcji kosztów.

Zastosowania uczenia maszynowego:

- oprogramowanie do rozpoznawania mowy:
 - automatyczne tłumaczenie
 - rozpoznawanie mowy ludzkiej
 - dyktowanie komputerowi
 - interfejsy użytkownika sterowane głosem
 - automatyzacja głosem czynności domowych
 - interaktywne biura obsługi
 - rozwój robotów
- automatyczna nawigacja i sterowanie:
 - kierowanie pojazdem (ALVINN)
 - odnajdywanie drogi w nieznanym środowisku
 - kierowanie statkiem kosmicznym (NASA Remote Agent)
 - automatyzacja systemów produkcji i wydobywania (przemysł, górnictwo)
- analiza i klasyfikacja danych:
 - systematyka obiektów astronomicznych (NASA Sky Survey)
 - rozpoznawanie chorób na podstawie symptomów
 - modelowanie i rozwijanie terapii lekowych
 - rozpoznawanie pisma na podstawie przykładów
 - klasyfikowanie danych do grup tematycznych według kryteriów
 - aproksymacja nieznanej funkcji na podstawie próbek
 - ustalanie zależności funkcyjnych w danych
 - przewidywanie trendów na rynkach finansowych na podstawie danych mikro i makro ekonomicznych
 - wykrywanie prania pieniędzy

Systemy ekspertowe

System ekspertowy jest to program, lub zestaw programów komputerowych wspomagający korzystanie z wiedzy i ułatwiający podejmowanie decyzji. Systemy ekspertowe mogą wspomagać bądź zastępować ludzkich ekspertów w danej dziedzinie, mogą dostarczać rad, zaleceń i diagnoz dotyczących problemów tej dziedziny.

Systemy ekspertowe składają się z co najmniej dwóch elementów:

- „silnika” – programu umożliwiającego zadawanie pytań i szukającego odpowiedzi na zadane pytania,
- bazy danych, na podstawie analizy której udzielane są odpowiedzi[5].

Zastosowania systemów ekspertowych:

- diagnozowanie chorób
- poszukiwanie złóż minerałów
- identyfikacja struktur molekularnych
- udzielanie porad prawnych
- diagnoza problemu (np. nieprawidłowego działania urządzenia)

- dokonywanie wycen i kalkulacji kosztów naprawy pojazdów przez zakłady ubezpieczeniowe.
- prognozowanie pogody
- sterowania robotami, automatycznymi pojazdami, rakietami, statkami kosmicznymi
- analiza notowań giełdowych

Zastosowania metod inteligencji obliczeniowej

<http://www.cs.put.poznan.pl/tpawlak/files/ZMIO/W01.pdf>

- Analiza danych, klasyfikacja i regresja
- Pamięć asocjacyjna (skojarzeniowa)
- Detekcja wzorców i klastrowanie
- Systemy autonomicznej kontroli
- Symulacja procesów populacyjnych

12. Inżynieria ontologii w przestrzeni Sieci Semantycznej.

<https://dataminingalapolonaise.files.wordpress.com/2009/10/ploug-oracle-semantic-technologies.pdf>

Ontologia

- logiczne teorie opisujące pewien aspekt rzeczywistości (dziedzinę)
- jest **strukturą**
 - określone **słownictwo** do opisu wycinka rzeczywistości
 - zbiór jawnie podanych **założeń** dotyczących **znaczenia** tego słownictwa (znaczenie zamierzone i jednoznacznie określone)
- opisuje formalną specyfikę wycinka rzeczywistości (określonej dziedziny)
 - **współdzielone** rozumienie **wiedzy** (wycinka rzeczywistości)
 - **formalny** model **dziedziny** możliwy do przetwarzania przez procesy automatyczne
- zawiera nie tylko suche fakty, ale też powiązania między nimi
- artefakty inżynierskie: reprezentacja pojęć, reprezentacja ontologii
- **konceptualizacja; jawna, formalna semantyka współdzielona**
- metadane zawierające bazę wiedzy dziedzinowej, której dotyczą opisywane dane (np. medycznej, chemicznej)

Struktura ontologii opisuje podstawowe elementy

- nazwy pojęć istotnych w dziedzinie
- wiedza podstawowa i ograniczenia

Ontologia - zastosowanie

- **udostępnienie zasobów WWW dla procesów automatycznych**
 - rozszerzenie istniejących tagów tagami semantycznymi - anotacja semantyczna, metadanymi
 - ontologie dostarczają znaczenia adnotacjom (specyfikacja dla automatów)
 - ustalenie założeń wstępnych jest standardem dla języków ontologii
 - wspólna składnia zanim będą dzielone znaczenia
 - sieć syntaktyczna bazuje na standardach (HTTP, HTML)
- **dodanie semantyki od "syntaktycznej" sieci WWW**
 - ontologie do nadania jednolitego znaczenia anotacjom (Dublin Core - pierwszy)
 - zbiór/słownik terminów
 - nowe terminy przez łączenie istniejących
 - znaczenie (semantyka) terminów określona w sposób formalny
 - relacje między terminami
- **dostarczanie narzędzi i serwisów użytecznych dla człowieka**
 - tworzenie i utrzymanie ontologii wysokiej jakości

- przechowywanie instancji klas ontologii - np. anotacje
- odpowiadania na zapytania o klasy i instancje
- integrowanie ontologii

Ontologia - przyczyny tworzenia

- uwspólnienie rozumienia struktury informacji, które można opisać między ludźmi, agentami software'owymi, ludźmi i agentami
- umożliwienie wykorzystania i dostosowania wiedzy dziedzinowej do różnych zadań

Inżynieria ontologii

- jakie założenia powinna spełniać dobra ontologia
- cechy dobrej metodologii budowania/tworzenia metodologii
- metodologie umożliwiające projektowanie ontologii oparte na współpracy twórców
- możliwości wykorzystania ontologii w innym zastosowaniu niż pierwotnie zakładano/zaprojektowano

Założenia

- **Rozszerzalność**
 - zaprojektowana w sposób ogólny, umożliwiający wykorzystanie i dostosowanie do różnych zadań
 - łatwo rozszerzalna - dodawanie nowych pojęć i relacji łatwe, oczywiste, bez konieczności zmiany istniejących założeń ontologii
 - proces monotoniczny - nie modyfikuje wcześniej utworzonej struktury
- **minimalna zależność od języka**
 - specyfika ontologii na poziomie wiedzy, niezależnie od języka zapisu ontologii
- **minimalne zaangażowanie ontologiczne**
 - przyjęcie minimalnych założeń wstępnych dotyczących dziedziny, które umożliwią inny użytkownikom wykorzystanie ontologii do własnych zastosowań

Metodologie inżynierii ontologii

- wspomagają proces tworzenia ontologii
- stanowią uporządkowany zbiór reguł tworzenia artefaktów konceptualnych (ontologii)
- instrukcja, niezbędne kroki do wykonania
- metodologie tworzenia ontologii
 - **niezakładające istnienia ontologii**
 - od podstaw
 - wydobywanie ontologii z tekstów
 - **zakładające istnienie ontologii**
 - przez modyfikowanie i rozwijanie istniejących ontologii
 - tworzenie związków między ontologiami
 - kooperacyjnego tworzenie ontologii
 - oceny (poprawności) ontologii

Metodologia ujednolicona

- **identyfikacja przeznaczenia ontologii i zakresu jej odbiorców**
 - rola ontologii podczas tworzenia i użytkowania systemu informatycznego
 - definiowanie testów kompetencji
 - wiedza od ekspertów dziedzinowych
 - analiza tekstów z dziedziny
- **budowa ontologii**
 - uchwycenie opisywanej rzeczywistości - identyfikacja terminów i pojęć istotnych dla modelowanej dziedziny (bottom-up, top-down, middle-out)
 - kodowanie ontologii - zapis ontologii w wybranym języku reprezentacji wiedzy
 - integracja z istniejącymi ontologiami
 - ewaluacja ontologii względem wymagań poprawności ontologicznej (założeń ontologii)
 - dokumentowanie założeń przyjętych przy tworzeniu ontologii i pojęć specyfikowanych w ontologii - możliwość jej ponownego użycia

Metodologia

1. **Ustalenie przeznaczenia i celu tworzenia ontologii**
 - bez ustalenia **celu** brak możliwości ustalenia **zakresu, wymagań i sposobu ewaluacji**
2. **Nieformalne wydobywanie wiedzy**
 - na przykład na karteczkach
 - określenie/zebranie terminów
 - uporządkowanie terminów - nieformalne
 - parafrazowanie, wyjaśnianie terminów do utworzenia nieformalnych definicji pojęć/konceptów
 - nieformalny diagram
3. **Opracowanie i weryfikacja wymagań i testów**
4. **Implementacja**
 - wytworzenie znormalizowanego schematu i szkieletu ontologii
 - implementacja prototypu, rejestrowanie intencji w postaci komentarzy i parafraz
 - pamiętać o **celu tworzenia** ontologii - weryfikacja i porównanie wyników prototypów z celami
 - rozszerzanie ontologii
 - sprawdzenie wydajności
 - udostępnianie
 - technologie wydobywania wiedzy z tekstów
 - przetwarzanie języka naturalnego
5. **Ewaluacja i zapewnienie jakości**
 - dołączenie testów zarządzania modyfikacjami
 - przygotowanie testów regresji i "próbek"
6. **Monitorowanie wykorzystania i ewoluowania ontologii**

Normalizacja

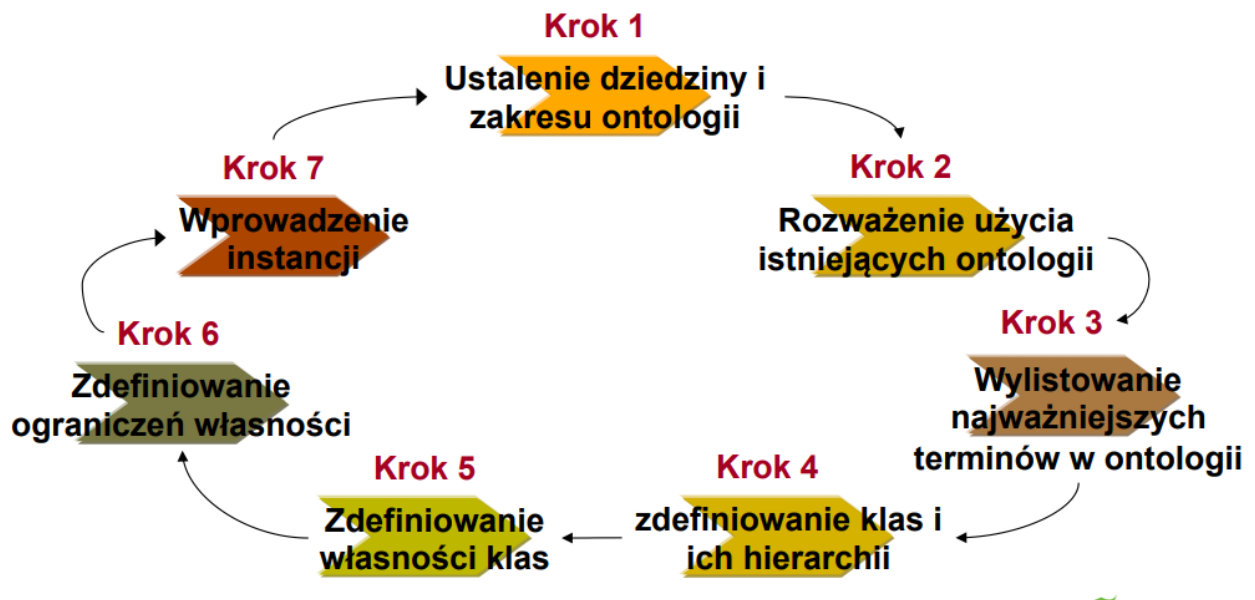
- początkowo ontologia ma postać drzewa

- przypisanie elementów atomowych do rozdzielnych drzew
- dodanie do drzew definicji i ograniczeń
- klasyfikator tworzy skierowany graf acykliczny

Tworzenie ontologii znormalizowanej

1. Identyfikacja elementów atomowych
2. Przypisanie elementów atomowych do rozdzielnych drzew
3. Identyfikacja relacji
4. Dodanie do drzew definicji i ograniczeń
5. Ustalenie zaklasyfikowanych kluczowych elementów
6. Użycie klasyfikatora - skierowany graf acykliczny
7. Wykonanie ustalonych testów dla pozostałych elementów ontologii

Dla elementów niejednoznacznych podczas normalizacji dodaje się komentarze



13. Koszty uwzględniane w kosztorysie projektu informatycznego.

Ze względu na złożoność realizacji projektów informatycznych (i konieczność uwzględnienia wielu nieplanowanych zdarzeń) szacowanie kosztu wytwarzania oprogramowania jest trudne (podobnie jak ustalanie harmonogramu).

Koszt realizacji projektu informatycznego składa się głównie z:

- kosztów osobowych (wynagrodzenie wykonawców i personelu wspomagającego)
- kosztów sprzętu i oprogramowania usługowego
- kosztów realizacji:
 - wyjazdy, szkolenia
 - utrzymanie pomieszczeń, materiały biurowe, itp.

Najtrudniejszym elementem szacowania kosztu realizacji przedsięwzięcia informatycznego jest określenie wydajności pracy (czyli np. ile zajmie realizacja określonego zadania przez określoną liczbę osób w określonych warunkach). W celu ułatwienia szacowania kosztu wprowadza się rozmaite metryki wydajności:

- metryki oparte na rozmiarze kodu (np. liczba linii kodu produkowanego miesięcznie przez pojedynczego programistę)
- metryki oparte na uzyskiwanej funkcjonalności (np. interakcjach z użytkownikiem, interakcjach z urządzeniami we/wy)

Metryki są bardzo szacunkowe i przybliżone, nie uwzględniają różnicowości i złożoności oprogramowania.

Szacowanie kosztu jest każdorazowo zadaniem złożonym i indywidualnym, można realizując go wspomagać się następującymi technikami:

1. modelowaniem algorytmicznym
2. ekspertyzami specjalistów
3. porównaniami z uprzednio realizowanymi projektami
4. ograniczeniami (szacowany jest koszt zależnie od ograniczeń i następnie przykrawa się wymagania i szczegóły realizacji do przewidzianego kosztu). Przykładem ograniczeń są np. warunki zamówienia lub stan firmy

14. Metody i narzędzia badania doświadczeń użytkownika.

Metody:

Badania User Experience stają się coraz bardziej popularne. Wyróżnić można badania makiet, layoutów, stron www, formularzy i wiele innych. Najważniejszy jest jednak dobór odpowiedniej metody.

Indywidualne wywiady pogłębione

Nazywane również IDI (skrót od angielskiej nazwy Individual in Depth Interview). Badanie polega na rozmowie moderatora z osobą badaną. Wywiad prowadzony jest na podstawie wcześniej przygotowanego scenariusza, który określa główne punkty rozmowy lub zawiera szczegółowy zestaw pytań, na które poszukujemy odpowiedzi. Badacz skupia się na zadawaniu pytań otwartych i dopytywaniu tak aby jak najlepiej poznać zachowanie respondenta. Czas trwania jednego wywiadu wynosi ok. 1-1,5 godziny.

Grupy fokusowe

Pomagają poznać opinie, emocje oraz motywacje zachowań. Dzięki zastosowaniu odpowiednich technik projekcyjnych umożliwiają zrozumienie nieujawnionych bezpośrednio postaw. Badanie polega na dyskusji prowadzonej i nadzorowanej przez moderatora (który posiłkuje się wcześniej przygotowanym scenariuszem odnoszącym się do kluczowych zagadnień). Ma ono na celu zaangażowanie uczestników w dyskusję i konfrontację ich opinii. W badaniu bierze udział zazwyczaj 6-8 osób. Odbывается ono w pomieszczeniu wyposażonym w lustro weneckie, a sam przebieg badania jest nagrywany.

Card Sorting

Niezwykle pomocne badanie na etapie tworzenia serwisu, a także późniejszej optymalizacji struktury informacji. Metoda ta polega na układaniu poszczególnych elementów w grupy. Jej głównym celem jest określenie jak użytkownicy postrzegają hierarchię elementów. W badaniu powinno brać udział przynajmniej 5 osób, a najlepiej 15. Najbardziej optymalna liczba kartek to 30-40 (badanie z większą liczbą kartek może być dla użytkowników trudne do realizacji z powodów czysto poznawczych).

Paper prototyping

Metoda służąca do weryfikacji pierwszych interfejsów, pozwalająca na iteracyjny cykl pracy nad produktem. Badanie polega na stworzeniu papierowej makiety interfejsu oraz przebadaniu jej z użytkownikiem – użytkownik wskazuje na kartce elementy, w które by kliknął poszukując danej informacji, a moderator podmienia rysunki. Badanie ułatwia wykrycie błędów funkcjonalnych systemu, nie skupia się jednak na layoutie produktu. Jest również dosyć czasochłonne w przygotowaniu.

Analizy eksperckie

Polegają na ocenie produktu przez eksperta UX (najlepiej z kilkuletnim doświadczeniem) pod kątem problemów produktu oraz ogólnej oceny jego użyteczności.

Możemy wyróżnić kilka metod:

- Analiza heurystyczna – metoda polegająca na analizie produktu pod kątem heurystyk, czyli ogólnych zasad użyteczności.
- Cognitive walkthrough – metoda polegająca na wcieleniu się eksperta w postać użytkownika, który wykonuje kluczowe zadania na stronie (z reguły 3- 5 scenariuszy). Celem badania jest połączenie punktu widzenia użytkownika oraz wiedzy eksperckiej.
- Lista kontrolna – metoda w której wykorzystywane są checklisty zasad działania produktu. Wynikiem analizy jest informacja o liczbie punktów czy też wartość procentowa. Czasami oprócz wskazań błędów audyt zawiera wiele dodatkowych uwag np. rekomendacje zmian, wskazanie dobrych praktyk konkurencji.

Eyetracking

Badanie polega na śledzeniu ruchu gałek ocznych użytkowników, za pomocą specjalnego urządzenia – eyetrackera. Badanym materiałem mogą być projekty graficzne, działające serwisy internetowe czy też materiały reklamowe. W badaniu powinno brać udział minimum 7 osób.

Badania laboratoryjne

Badanie polega na wykonywaniu przez użytkowników wcześniej zdefiniowanych scenariuszy. Przebiega ono z reguły w sali wyposażonej w lustro weneckie – w jednym pomieszczeniu znajduje się moderator wraz z osobą badaną, w drugiej zaś obserwatorzy. Zazwyczaj w badaniu wykorzystywany jest program nagrywający pulpit i twarz respondentów. Badanie ma na celu wykrycie funkcjonalnych błędów produktu. Testować również można layouty oraz strony konkurencyjne. Badaniu często uczestniczy wywiad pogłębiony, który realizowany jest po wykonaniu każdego zadania, lub też podsumowując na sam koniec. Z reguły bierze w nim udział 5-8 osób.

Zdalne badania z użytkownikami

Badania przeprowadzane drogą internetową (bez fizycznego kontaktu z respondentem). Polegają na obserwacji zachowań użytkowników, którzy mają za zadanie wykonać określone zadania na stronie. Główną zaletą tego typu badań są naturalne warunki ich realizacji- testerzy wykonują je w domu, bądź w pracy. Dodatkowo można wspomnieć o szybkim czasie realizacji i stosunkowo niskich kosztach.

Testy AB

Metoda badawcza polegająca na porównaniu dwóch wersji strony internetowej, w celu wybrania lepszej. Badaniu może podlegać różnica względem jednego lub kilku elementów. W teście powinno uczestniczyć minimum 50 respondentów na wersję.

Clicktracking

Służy do śledzenia aktywności użytkowników na stronach. Metoda polega na pomiarach kliknięć użytkowników co do piksela (brane są pod uwagę informacje o rozdzielczości, wielkości okna przeglądarki, miejsca kliknięcia). Badanie pozwala sprawdzić, które miejsca na stronie są klikane najczęściej (z których opcji korzystają użytkownicy, które miejsca uważają za klikalne a np. nie są hiperłączami).

Narzędzia:

Google Analytics

Darmowe narzędzie dostarczające wielu cennych informacji o użytkownikach – już podstawowe raporty pozwolą przyjrzeć się ruchowi w serwisie i dowiedzieć się kim są i skąd przychodzą. Dzięki śledzeniu zdarzeń i ustawieniu celów będzie możliwe monitorowanie skuteczności prowadzonych kampanii i stopień realizacji postawionych przed serwisem zadań. Wystarczy, że kod strony uzupełnimy o kod śledzący, który umożliwi zbieranie danych. Ich analiza wskaże obszary wymagające optymalizacji i będzie pierwszym krokiem do poznania użytkowników.

Morae

Rozbudowany niezbędnik badacza. W skład pakietu wchodzi aż trzy narzędzia – Recorder, Observer i Manager, dzięki którym rejestrujemy, przetransmitujemy, a także zanalizujemy i zedytujemy nagrania z badań z użytkownikami. Sprawdzą się zarówno w przypadku testów użyteczności prowadzonych w laboratorium (np. na prototypie aplikacji, na urządzeniach mobilnych), jak i warsztatów czy fokusów.

UserZoom

Wielofunkcyjna platforma do przeprowadzania zdalnych testów użyteczności. Dostarcza szczegółowych informacji o zachowaniach i opiniach odbiorców naszego produktu. Rekrutacja respondentów odbywa się w serwisie, za pomocą specjalnego modułu – wykonują oni zadania na swoim urządzeniu, podczas naturalnego korzystania ze strony. Połączenie badań jakościowych z ilościowymi pozwala uchwycić istotę problemu i naprowadzić na drogę jego naprawy.

Personapp

Darmowa aplikacja do tworzenia person – przykładów modelowych użytkowników grupy docelowej. Do dyspozycji mamy podstawowy schemat (zawierający metryczkę, zachowania, potrzeby i cele) i wybór szkicowych portretów. Uzupełniony szablon możemy następnie wydrukować lub udostępnić członkom zespołu – przydatne w momencie, gdy zależy nam na czasie.

Balsamiq

Balsamiq to intuicyjne narzędzie do tworzenia szkiców interfejsu i prototypowania podstawowych interakcji. Aplikacja oferuje bogatą bibliotekę gotowych elementów – także dla iOS. Przeciągając wybrane elementy z biblioteki ekspresowo stworzysz makiety niskiej szczegółowości – wizualizując pomysły np. podczas spotkania z klientem.

15. Narzędzia analizy dużych danych (ang. Big Data).

Dzisiejszy rynek jest zalany szeroką gamą narzędzi Big Data. Zapewniają efektywność kosztową, lepsze zarządzanie czasem w zadaniach analitycznych danych.

Apache Hadoop:

Biblioteka oprogramowania Apache Hadoop to duże środowisko danych. Umożliwia rozproszone przetwarzanie dużych zestawów danych między klastrami komputerów. Została zaprojektowana do skalowania z pojedynczych serwerów do tysięcy komputerów.

Funkcje:

- Ulepszenia uwierzytelniania podczas korzystania z serwera proxy HTTP
- Specyfikacja działania systemu zgodnego z systemem plików Hadoop
- Obsługa rozszerzonych atrybutów systemu plików POSIX
- Oferuje solidny ekosystem, który jest dobrze dostosowany do potrzeb analitycznych dewelopera
- Zapewnia elastyczność w przetwarzaniu danych
- Pozwala na szybsze przetwarzanie danych

HPCC:

HPCC to duże narzędzie do opracowywania danych opracowane przez LexisNexis Risk Solution. Dostarczona na jednej platformie, jednej architekturze i jednym języku programowania do przetwarzania danych.

Funkcje:

- Wysoce wydajna realizacja dużych zadań przy znacznie mniejszej ilości kodu.
- Oferuje wysoką redundancję i dostępność
- Może być używany do złożonego przetwarzania danych w klastrze Thora
- Graficzne IDE dla uproszczenia programowania, testowania i debugowania
- Automatycznie optymalizuje kod do przetwarzania równoległego
- Zapewnia lepszą skalowalność i wydajność
- Kod ECL kompiluje się do zoptymalizowanego C ++, a także może rozszerzać przy użyciu bibliotek C ++

Storm:

Storm to darmowy i otwarty system obliczeń big data. Oferuje rozproszony w czasie rzeczywistym, odporny na uszkodzenia system przetwarzania. Dzięki możliwościom obliczeń w czasie rzeczywistym.

Funkcje:

- Testowano go jako przetwarzanie miliona wiadomości o wielkości 100 bajtów na sekundę na węzeł
- Używa równoległych obliczeń, które przebiegają w klastrze maszyn
- Uruchomi się automatycznie ponownie w przypadku śmierci węzła. Pracownik zostanie uruchomiony ponownie w innym węźle
- Storm gwarantuje, że każda jednostka danych zostanie przetworzona co najmniej raz lub dokładnie jeden raz
- Po wdrożeniu Storm jest z pewnością najłatwiejszym narzędziem do analizy Bigdata

Qubole Data:

Qubole Data to autonomiczna platforma zarządzania danymi. Jest to samo-zarządzane, samopoprawiające narzędzie, które pozwala zespołowi skoncentrować się na wynikach biznesowych.

Funkcje:

- Jedna platforma dla każdego przypadku użycia
- Silniki Open-source, zoptymalizowane pod kątem chmury
- Kompleksowe bezpieczeństwo, zarządzanie i zgodność
- Udostępnia alerty, informacje i zalecenia, które można wykorzystać, aby zoptymalizować niezawodność, wydajność i koszty

Apache Cassandra:

Baza danych Apache Cassandra jest obecnie powszechnie wykorzystywana do efektywnego zarządzania dużymi ilościami danych.

Funkcje:

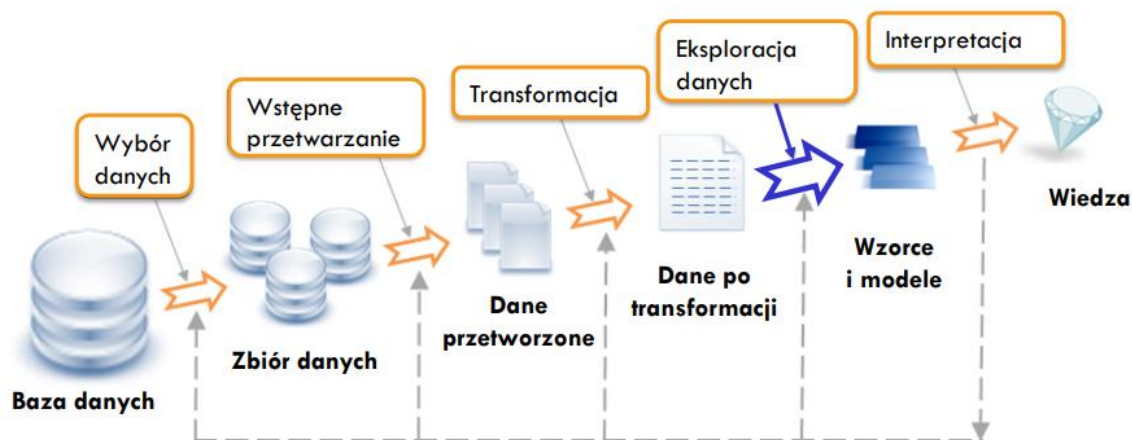
- Obsługa replikacji w wielu centrach danych dzięki mniejszemu opóźnieniu dla użytkowników
- Dane są automatycznie replikowane do wielu węzłów pod kątem tolerancji błędów
- Jest najbardziej odpowiednia dla aplikacji, które nie mogą pozwolić sobie na utratę danych, nawet gdy całe centrum danych jest wyłączone

16. Podstawowe metody analizy dużych baz danych.

<http://www.ploug.org.pl/wp-content/uploads/ploug-konferencja-05-7.pdf>

<http://aragorn.pb.bialystok.pl/~mczajk/ml/w1a.pdf>

https://www.statsoft.pl/textbook/stathome_stat.html?https%3A%2F%2Fwww.statsoft.pl%2Ftextbook%2Fstdatmin.html



Etapy procesu odnoszą się do przygotowania danych, wyboru danych do eksploracji, czyszczenia danych, definiowania dodatkowej wiedzy przedmiotowej, interpretacji wyników eksploracji i ich wizualizacji.

Metody eksploracji danych można podzielić, bardzo ogólnie, na 6 zasadniczych klas:

- **Odkrywanie asocjacji**

Najszersza klasa metod obejmująca, najogólniej, odkrywanie różnego rodzaju nieznanych zależności w bazie danych. Metody te obejmują głównie odkrywanie asocjacji pomiędzy obiektami. Generalnie, odkrywane zależności posiadają pewne miary statystyczne określające ich wsparcie i ufność.

- **Klastrowanie (grupowanie)**

Celem tych metod jest znajdowanie skończonego zbioru klas obiektów (klastrow) w bazie danych posiadających podobne cechy. Liczba klastrow jest nieznana, stąd, proces klastrowania przebiega, najczęściej, w dwóch cyklach: cykl zewnętrzny przebiega po liczbie możliwych klastrow, cykl wewnętrzny próbuje znaleźć optymalny podział obiektów pomiędzy klastry.

- **Odkrywanie wzorców sekwencji**

Odkrywanie czasowych wzorców zachowań, np. znajdowanie sekwencji notowań giełdowych, zachowań klientów ubezpieczalni, klientów supermarketów.

- **Odkrywanie klasyfikacji**

Celem tych metod jest znajdowanie zależności pomiędzy klasyfikacją obiektów (klasyfikacja naturalna bądź wprowadzona przez eksperta) a ich charakterystyką. Zastosowanie: charakterystyka pacjentów, klientów kart kredytowych, pożyczkobiorców.

- **Odkrywanie podobieństw w przebiegach czasowych**

Znajdowanie podobieństw w przebiegach czasowych opisujących określone procesy.

- **Wykrywanie zmian i odchyłeń**

Znajdowanie różnic pomiędzy aktualnymi a oczekiwanymi wartościami danych: znajdowanie anomalnych zachowań klientów ubezpieczalni, klientów kart kredytowych, klientów firm telekomunikacyjnych.

17. Podstawowe metody i narzędzia inteligencji biznesowej.

<https://fi.nancesonline.com/15-best-business-intelligence-tools-small-big-business/>

Inteligencja biznesowa – proces przekształcania danych w informacje, a informacji w wiedzę, która może być wykorzystana do zwiększenia konkurencyjności przedsiębiorstwa. W rozumieniu czysto biznesowym BI można także definiować jako kombinację architektury systemu, aplikacji oraz baz danych, które razem umożliwiają prowadzone w czasie rzeczywistym analizy i przekształcenia, dostarczające potrzebną informację i wiedzę biznesowi.

Efektywne eksploatowanie narzędzi BI jest mocno uzależnione od utworzenia hurtowni danych, która pozwala na ujednolicenie i powiązanie danych zgromadzonych z różnorodnych systemów informatycznych przedsiębiorstwa. Utworzenie hurtowni danych zwalnia systemy transakcyjne od tworzenia raportów i umożliwia równoczesne korzystanie z różnych systemów BI.

Koncepcja jest następująca: system BI generuje standardowe raporty lub wylicza kluczowe wskaźniki efektywności działania przedsiębiorstwa (key performance indicators), na podstawie których stawia się hipotezy, po czym weryfikuje się je poprzez wykonywanie szczegółowych „przekrojów” danych. Do tego służą różnego rodzaju narzędzia analityczne (np. OLAP, eksploracja danych).

Do technik, jakimi posługuje się BI można zaliczyć m.in. **eksplorację danych, eksplorację procesów, systemy ekspertowe, sieci neuronowe i algorytmy genetyczne.**

Narzędzia:

<https://selecthub.com/business-intelligence/key-types-business-intelligence-tools/>

Google Analytics

Tableau - serwis webowy, prosty dla użytkownika. It's simple enough that non-technical users can easily create customized dashboards and other reports to find valuable insights.

Qlik - This powerful program offers fully interactive data operation, which allows users to explore data at all times during input, processing and output

Microsoft Power BI - interactive visualizations, predictive analytics, mobile access, export of analytics reports, big data integration and storyboarding

18. Rodzaje diagramów projektowych - główne elementy strukturalne diagramu i jego przeznaczenie.

Diagramy UML

<https://www.michalwolski.pl/diagramy-uml/>

Jednym z najistotniejszych elementów procesu twórczego oprogramowania jest komunikacja. Przełożenie skomplikowanych reguł biznesowych z języka biznesu na język informatyki to niewątpliwie trudne działanie, które trzeba wykonać budując lub modyfikując oprogramowanie. Język naturalny (mowa, pismo), którym posługujemy się by zobrazować rzeczywistość może okazać się zbyt skomplikowany i niejednoznaczny na poszczególnych etapach opisywania powstającego systemu.

Niezbędny jest więc taki sposób opisu, który byłby jednakowo interpretowany i zrozumiały dla wszystkich członków zespołu projektowego. Standardem w tej dziedzinie stał się język UML (Unified Modelling Language) – graficzny system wizualizacji, specyfikowania oraz dokumentowania składników systemów informatycznych. UML to notacja umożliwiająca zaprezentowanie systemu w sposób graficzny, w postaci diagramów. Modele zapisane w języku UML prezentują system od ogółu do szczegółu, umożliwiając oglądanie modelu systemu z wybraną w danym momencie szczegółowością.

Diagram przypadków użycia

Diagram przypadków użycia (ang. use case diagram) jest diagramem, który przedstawia funkcjonalność systemu wraz z jego otoczeniem.

Diagramy przypadków użycia pozwalają na graficzne zaprezentowanie własności systemu tak,

jak są one widziane po stronie użytkownika.

Diagramy przypadków użycia służą do zobrazowania usług, które są widoczne z zewnątrz systemu.

Diagram przypadków użycia, mimo iż jest zbudowany z kilku elementów, odgrywa najważniejszą rolę w procesie projektowania systemu; opisuje bowiem wymagania funkcjonalne, jakim system musi sprostać, i otoczenie, w którym się znajduje. Diagram ten jest agregatem funkcji usług, które wykonuje system. Poza specyfikacją, diagram ten pozwala na identyfikację funkcjonalności, weryfikację postępów w modelowaniu i implementacji, a także wspomaga komunikację pomiędzy uczestnikami projektu.

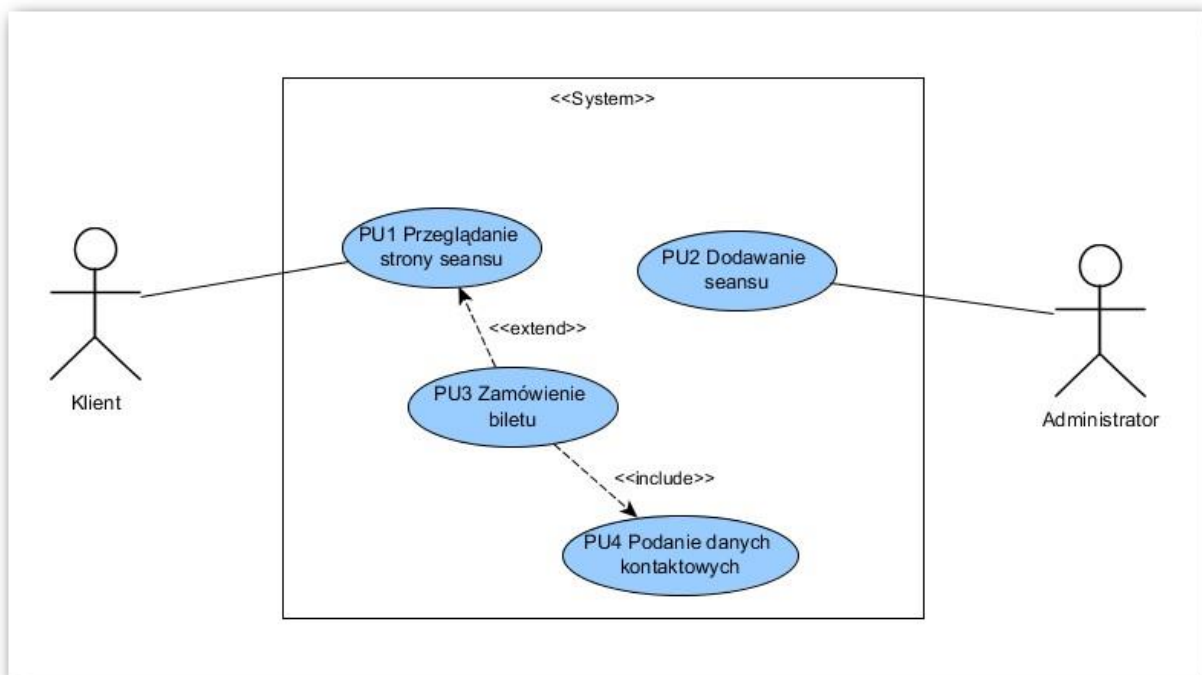


Diagram pakietów

Diagram pakietów (ang. package diagram) jest strukturalnym diagramem, który prezentuje pakiety i relacje zachodzące pomiędzy nimi. Diagram pakietów służy do modelowania agregatów bytów, jakimi są pakiety. Diagramy pakietów pozwalają na modelowanie systemu na wysokim stopniu abstrakcji, gdyż pakiety reprezentują ogromną liczbę klas, interfejsów, diagramów i innych bytów – pozwala to wyeksponować tylko zasadnicze funkcje systemu.

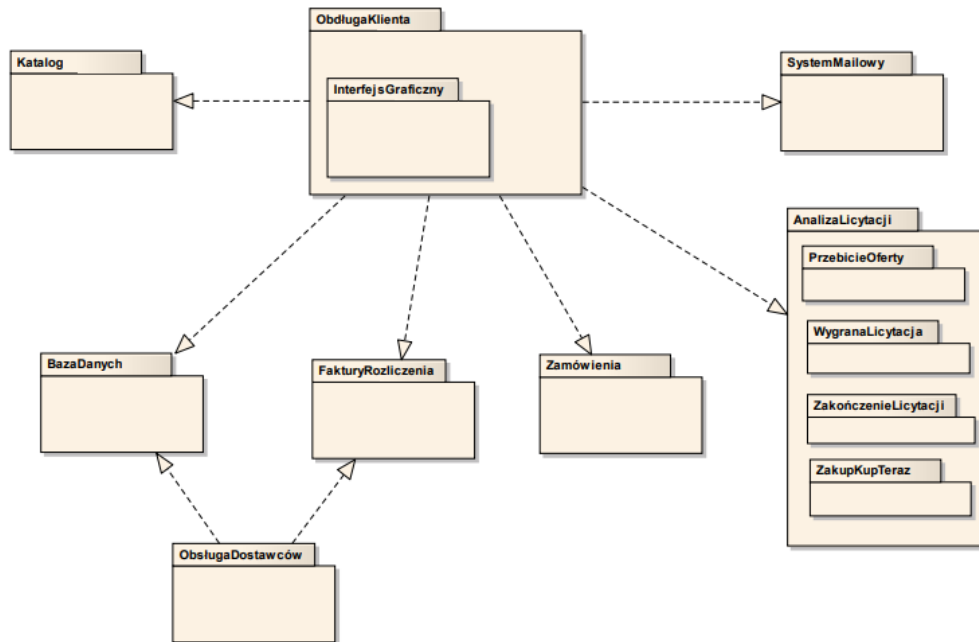


Diagram klas

Diagram klas obrazuje pewien zbiór klas, interfejsów i kooperacji oraz związki między nimi. Jest on grafem złożonym z wierzchołków (klas, interfejsów, kooperacji) i łuków (reprezentowanych przez relacje). Diagram klas stanowi opis statyki systemu, który uwypukla związki między klasami, pomijając pozostałe charakterystyki. Najsilniej prezentuje on więc strukturę systemu, stanowiąc podstawę dla jego konstrukcji. W modelowaniu złożonych systemów nie mamy obowiązku przedstawiania ich struktury na jednym diagramie. Powinniśmy pamiętać o tym, że złożenie wszystkich diagramów, a właściwie ich elementów i relacji, stanowi kompletny model. Możemy zatem przyjąć, że podzbiory zbioru klas użyte na diagramach klas są wybierane celowo i stanowią wynik decyzji zarówno analitycznych, jak i projektowych. Tak na przykład, diagramy klas stanowiące wynik decyzji analitycznych tworzą tzw. widoki klas biorących udział w realizacji danego przypadku użycia (ang. view of participating class – VOPC) i powiązania między nimi.

Zasadniczo jednak diagramy klas służą do zobrazowania statycznych aspektów perspektywy projektowej, w której bierze się pod uwagę wymagania funkcjonalne systemu – usługi, jakie system powinien udostępniać swoim użytkownikom.

Diagram sekwencji (ang. sequence diagram) służy do prezentowania interakcji pomiędzy obiektami wraz z uwzględnieniem w czasie komunikatów, jakie są przesyłane pomiędzy nimi. Na diagramie sekwencji obiekty ułożone są wzdłuż osi X, a komunikaty przesyłane są wzdłuż osi Y. Zasadniczym zastosowaniem diagramów sekwencji jest modelowanie zachowania systemu w kontekście scenariuszy przypadków użycia. Diagramy sekwencji pozwalają uzyskać odpowiedź na pytanie, jak w czasie przebiega komunikacja pomiędzy obiektami. Dodatkowo diagramy sekwencji stanowią podstawową technikę modelowania zachowania systemu, które składa się na realizację przypadku użycia.

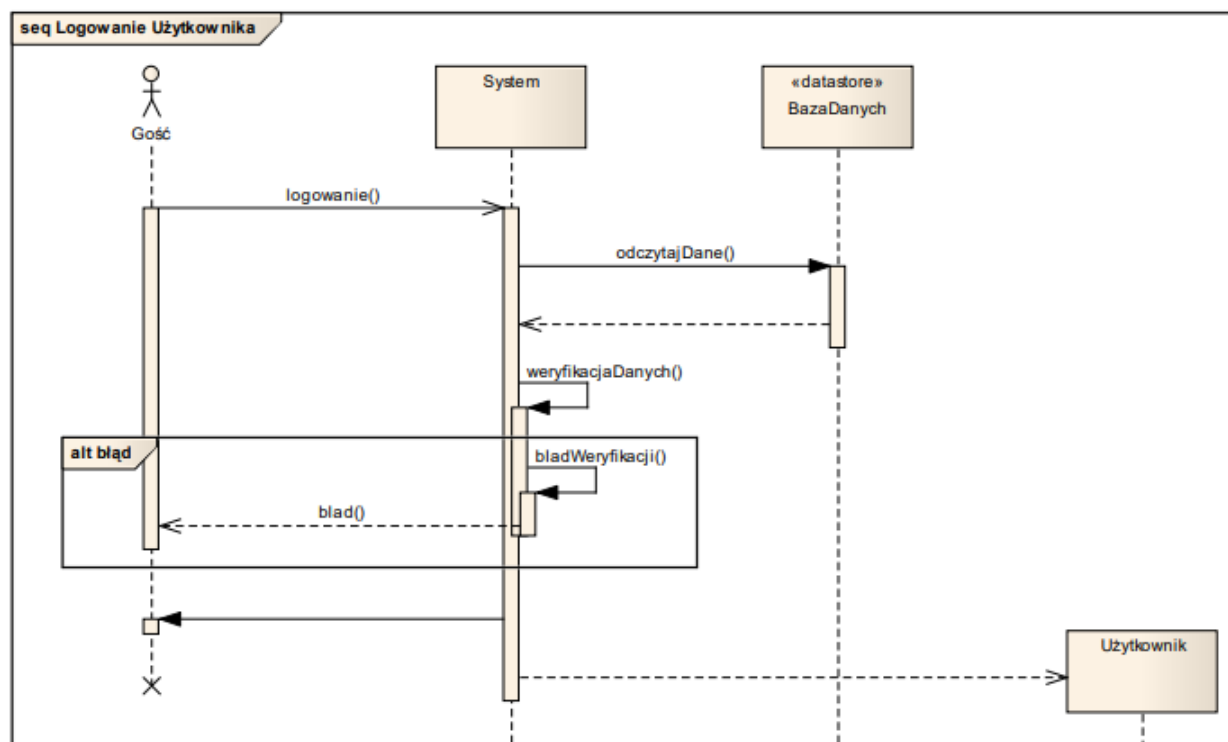


Diagram komponentów

Diagram komponentów (ang. component diagram) służy do ilustracji organizacji i zależności pomiędzy komponentami. Diagram komponentów prezentuje system na wyższym poziomie abstrakcji niż diagram klas, gdyż każdy z komponentów może być implementacją jednej lub większej liczby klas. Diagramy komponentów służą do określania szczegółów niezbędnych do budowy systemu.

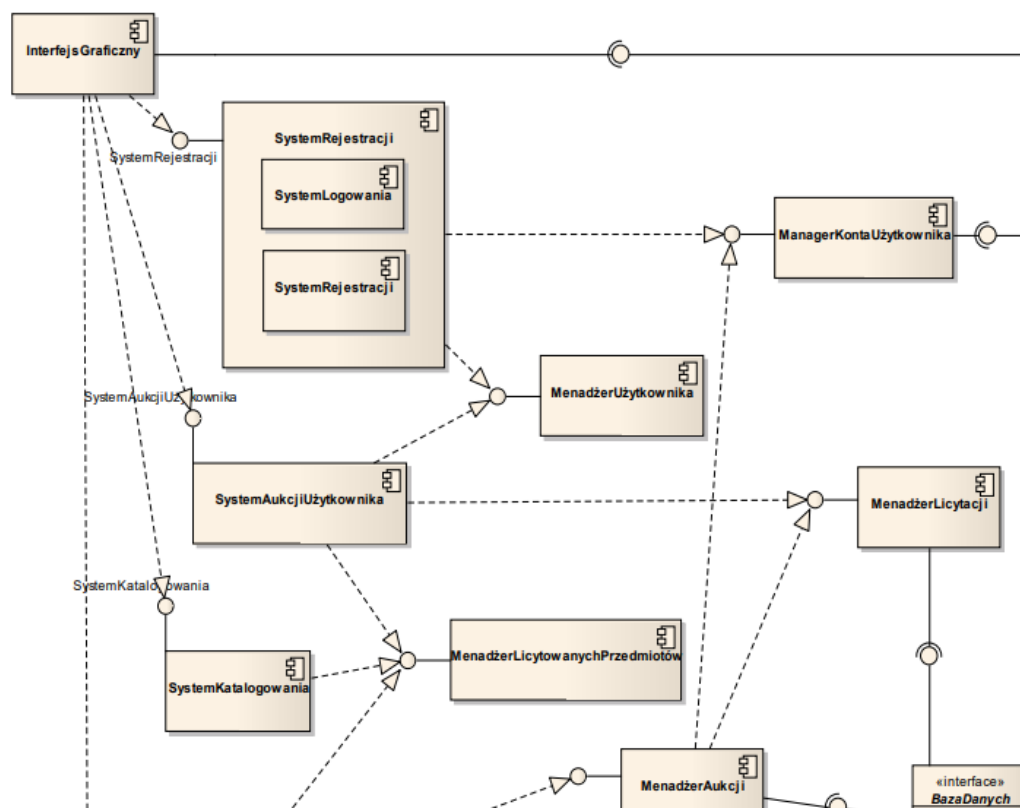


Diagram przypadków użycia	Identyfikacja kategorii użytkowników oraz sposobów używania przez nich systemu
Diagram klas	Modelowanie klas obiektów i ich wzajemnych relacji
Diagram czynności	Modelowanie procesów biznesowych, scenariuszy przypadków użycia lub algorytmów
Diagram stanów	Modelowanie historii życia obiektu – jego stanów i możliwych przejść między stanami
Diagram komponentów	Modelowanie fizycznych składników oprogramowania, ich zależności i interfejsów
Diagram rozmieszczenia (diagram wdrożenia)	Modelowanie konfiguracji sprzętowych i programowych komponentów systemu
Diagram sekwencji	Modelowanie czasowej sekwencji wymiany komunikatów podczas współpracy obiektów, pakietów lub komponentów
Diagram interakcji	Modelowanie przepływu sterowania w procesie biznesowym lub systemie

Diagram obiektów	Modelowanie chwilowej konfiguracji obiektów oprogramowania
-------------------------	--

19. Rodzaje dokumentacji systemu informatycznego, tworzonej w trakcie projektowania i realizacji systemu.

Dokumentacja programu – ogół dokumentacji technicznej i dokumentacji użytkownika stworzonej dla określonego programu komputerowego przez jego twórców.

W jej skład wchodzi:

- dokumentacja użytkownika to opis programu przeznaczony dla jego użytkownika. Składają się na nią np. pliki pomocy, ogólne informacje o programie i jego sposobie obsługi.
- dokumentacja techniczna jest przeznaczona dla osób, które mogą potrzebować modyfikować program. Zawiera dokładny opis metody działania programu, algorytmów w nim zastosowanych, rozmieszczenia i sposobu działania poszczególnych komponentów itp. Ze względu na swoją naturę jest ona przeznaczona dla programistów, a dla zwykłego użytkownika właściwie niezrozumiała.

Tradycyjnie pisanie dokumentacji jest czynnością wykonywaną przez programistów niechętnie, stąd powstają systemy automatycznego generowania dokumentacji technicznej wprost z kodu źródłowego, np. doxygen.

Dokumentacja programu powinna w swojej podstawie zawierać dokumenty piśmiennicze (w języku naturalnym, w formie zrozumiałej dla człowieka, z ang. human readable), choć w zależności od wymagań i rodzaju dokumentacji wskazane może być posiłkowanie się dokumentami niepiśmienniczymi, takimi jak fragmenty kodów źródłowych, wykresy, graficzne reprezentacje algorytmów, zdjęcia interfejsu użytkownika, diagramy przepływu, opisy UML czy XML).

3.1. Kategorie Dokumentacji Systemu

- 3.1.1. „**Dokumentacja użytkowa**” zawierająca dokumentację dla użytkowników zewnętrznych, użytkowników wewnętrznych oraz administratora merytorycznego,
- 3.1.2. „**Dokumentacja administracyjna**” zawierająca wszystkie niezbędne do utrzymania oprogramowania informacje, w szczególności opis instalacji i deinstalacji Systemu, opis archiwizacji i odtworzenia Systemu, typowych czynności administracyjnych,
- 3.1.3. „**Dokumentacja analityczno-projektowa**” przedstawiająca zasady konstrukcji oraz opis funkcjonalny i techniczny Systemu oraz jego poszczególnych modułów składowych,
- 3.1.4. „**Architektoniczny Model Systemu**” w Repozytorium Architektonicznym - podstawa do utworzenia dokumentacji Systemu.
- 3.1.5. „**Dokumentacja wspierająca proces wytwórczy**” wyjaśniająca zasady wytwarzania oprogramowania Systemu, w szczególności zasady zarządzania konfiguracją czy budowania kodu źródłowego do postaci wykonywalnej,
- 3.1.6. „**Dokumentacja Kodu Źródłowego**” wyjaśniająca działanie kodu źródłowego, a w zasadniczej części stanowiąca jego komentarze,
- 3.1.7. „**Dokumentacja zapewnienia jakości**” przedstawiająca zarówno podejście do zapewnienia jakości w ramach rozwoju Systemu, jak i plany, specyfikacje oraz raporty z testów.

http://firma.um.warszawa.pl/wp-content/uploads/2018/11/Za%C5%82%C4%85cznik_nr_4-Standard-dokumentacji-systemu-informatycznego.pdf

20. Rola i zadania wydawcy w procesie tworzenia gier komputerowych.

Wydawca gier komputerowych – firma publikująca gry komputerowe, tworzone przez własne bądź wybrane studia producentów gier komputerowych.

Wydawcy gier komputerowych są odpowiedzialni za tworzenie, marketing (włączając w to badania rynku) oraz wszystkie aspekty reklamy.

Zwykle to wydawca finansuje rozwój gry, czasem płacąc producentowi gry (czasem dzięki tej umowie, producent otrzymuje pracowników do pomocy w grze, nazywanych studiem).

Więksi wydawcy zajmują się także dystrybucją gier, które wydają, podczas gdy mniejsi wydawcy wynajmują firmy dystrybutorskie (lub większych wydawców), do dystrybucji swoich gier.

Inne funkcje, których zwykle podejmuje się wydawca to wybór i zapłata za licencje, na podstawie których można stworzyć grę, zapłata za lokalizację gry, stworzenie podręcznika użytkownika oraz tworzenie elementów graficznych, takich jak loga, czy wygląd pudełka gry. Więksi wydawcy mogą także spróbować zwiększyć wydajność wszystkich wewnętrznych i zewnętrznych wydziałów rozbudowy przez dostarczanie takich usług, jak projektowanie dźwięku.

Ponieważ zwykle wydawcy finansują tworzenie gry, często próbują poradzić sobie z jego ryzykiem radzą się producentów lub kierowników projektu, by monitorowali postęp tworzenia produktu i asystowali przy nim, jeżeli będzie to konieczne. Większość zewnętrznych wydawców gier opłaca swoje dzieła w odpowiednich etapach produkcji, nazywanych kamieniami milowymi.

Ryzyko biznesowe:

Jak każdy biznes, wydawanie gier również może być ryzykowne:

- Średnio około połowa rocznych dochodów ze sprzedaży gier komputerowych wiąże się z okresem Bożego Narodzenia, co powoduje coroczne konkurowanie wszystkich wydawców w każdym gatunku gier.
- Poślizg w wydaniu projektu jest bardzo częstym powodem niezbyt sprecyzowanych dat wydanych przez producenta. Każdy wydawca zaliczył tzw. "falstart", gdy ekipa pracująca nad projektem zapewnia firmę, że pracę nad nim skończą się w dniu x, a promowanie produktu rozpoczyna się w dniach zbliżonych do tej daty, włączając w to reklamowanie produktu, po czym, producent informuje, że gra będzie gotowa w dniu x plus cztery miesiące. Gdy gra w końcu jest gotowa i pojawia się na rynku, klienci nie są już tak podekscytowani produktem, co skutkuje mniejszymi zyskami firmy. Problemy te są jeszcze większe, jeżeli gra wyczekiwana jest w okresie Bożego Narodzenia, a wydanie gry zostanie przeniesione na nowy rok.

- Podczas publikacji gry na konsole wydawcy biorą na siebie ciężar wielkiego ryzyka. Wszyscy znaczący wytwórcy konsoli wymagali lub wymagają od wydawców zapłaty za każdą wytworzoną grę, która może być uruchomiona na ich konsolach. Opłata musi zostać uiszczona w czasie produkcji gry. Opłata ta przechodzi na cenę gry, przez co gry dostępne na konsole są droższe niż te same gry wydawane na komputery osobiste.

Obowiązki wydawcy gier mogą obejmować negocjowanie umów dystrybucyjnych, projektowanie kampanii reklamowych, przeprowadzanie badań rynkowych i usuwanie wszelkich problemów związanych z licencjami w grze.

21. Semantyczne wyszukiwania informacji w sieci Web.

<https://makoblog.com/pl/czym-jest-i-jak-dziala-semantic-search/>

<https://www.sunrisesystem.pl/blog/2207-wyszukiwanie-semantyczne-zapytaj-google.html>

Najlepiej to przedstawić na przykładzie wyszukiwarki Googla. Na początku do wyszukiwania była brana fraza i słowa kluczowe. Później został wprowadzony **algorytm Koliber**, dzięki któremu trafność wyników wyszukiwania znacznie się poprawiła. Koliber miał bowiem na celu lepsze zrozumienie intencji użytkownika. W praktyce oznaczało to bardziej znaturalizowane wyszukiwanie i zamiast zestawu fraz, można było wprowadzać pełne zapytania np. „Gdzie w Poznaniu jest sklep z butami Adidasa?”.

Algorytm był niejako dopełnieniem innej funkcjonalności, stanowiącej podwaliny pod współczesne semantyczne wyszukiwanie na masową skalę (istniały mniej znane wyszukiwarki semantyczne jak np. Wolfram Alfa, True Knowledge, Yebol, Google Squared). Chodzi o Graf Wiedzy.

Nowa funkcja wyszukiwarki Google sprawiła, że wyszukiwanie przestało koncentrować się wyłącznie na frazach. **Zadaniem Grafu Wiedzy było znalezienie konkretnej informacji na zadane pytanie** (a pytań może być wiele i to zadanych na różne sposoby). **Zapewniał także użytkownikom możliwość poruszania się po tematach powiązanych.**

Co odróżniało Graf Wiedzy od dotychczasowych wyszukiwań?

- znalezienie najtrafniejszego podsumowania – umożliwia zebranie najważniejszych informacji w jednym miejscu. Zebrane dane są zaprezentowane w pewnej strukturze, np. w przypadku osoby mamy informacje kiedy się urodziła i zmarła, kim była, jakie są jej dzieła lub dokonania, itp.
- znalezienie konkretnej rzeczy – wiązało się to z wieloznacznością słów, w niektórych bowiem przypadkach pod jednym hasłem może kryć się kilka znaczeń, tak przykładowo po wpisaniu „Chopin” w wyszukiwarkę może pojawić nam się w wynikach zarówno kompozytor, jak i lotnisko w Warszawie imienia tego właśnie kompozytora.
- rozszerzenie oraz pogłębienie poszukiwań – Graf Wiedzy pozwolił na zgłębienie wiedzy na dany temat poprzez odkrycie nowych powiązań z innymi tematami, np. możemy dowiedzieć się, że w Polsce znajduje się Muzeum Chopina w Warszawie, że organizowany jest Festiwal „Chopin i jego Europa”, itp.

Wprowadzenie algorytmu Koliber oraz Grafu Wiedzy było oznaką: po pierwsze – tego, w którą stronę zmierza Google w zakresie wyszukiwarki, po drugie - tego, czego oczekują od wyszukiwarki odbiorcy. A oczekują uzyskania konkretnej odpowiedzi na naturalnie sformułowane zapytania. To właśnie zapewnia wyszukiwarka semantyczna.

Jej zadaniem jest przeszukiwanie zawartości strony internetowej przy jednoczesnym ciągłym interpretowaniu ich treści poprzez semantyczną i gramatyczną analizę języka naturalnego tworzącego stronę. Ponadto wyszukiwarki semantyczne „uczą się” nowych powiązań, nowych relacji, dzięki czemu w efekcie mają dostarczać precyzyjniejsze wyniki.

Obecnie wyszukiwarka Google do tworzenia powiązań między informacjami wykorzystuje Wikipedię oraz otwarte lub wykupione bazy danych (jak Freebase). Docelowo użytkownicy mają poprzez swoją aktywność (najczęściej odwiedzane strony, zainteresowania, czytane artykuły, poszukiwane wydarzenia, itp.) budować sieć powiązań – właściwą dla siebie.

22. Standardy opisu treści w Sieci Semantycznej.

Sieć semantyczna

- **idea**, żeby zdefiniować i powiązać dane możliwe do automatycznego przetwarzania przez aplikacje
- **inicjatywa**, której celem jest rozszerzenie aktualnej sieci WWW i wyposażenie jej w powszechne i automatyczne zasoby
- Aktualnie nie jest ona realizowalna, jednak można wykonywać kroki w celu jej realizacji
- stworzenie “Sieci Zaufania” (**Web of Trust**), dostarczającej platformy pozwalającej na współdzielenie i przetwarzanie danych przez automatyczne narzędzia i przez ludzi
- zautomatyzowane przetwarzanie zawartości znaczeniowej dokumentów (np. wyszukiwanie na podstawie znaczenia, a nie po słowach kluczowych)

Koncepcja sieci semantycznej

- do funkcjonowania sieci semantycznej trzeba dostarczyć komputerom **źródeł informacji o ściśle określonej strukturze i reguł inferencyjnych**, które pozwolą na automatyczne wyciąganie wniosków
- rozpoznanie informacji - dotarcie po odnośnikach do **dokumentów z definicjami pojęć** używanych na stronie i **reguł wnioskowania**
- taka architektura ułatwia rozwój zautomatyzowanych serwisów i **agentów** do wykonywania **złożonych zadań**
- Sieć semantyczna ma korzystać z istniejącej infrastruktury technicznej internetu, zmienia się jedynie sposób przesyłania danych

Anotacja semantyczna - rodzaj metadanych, które dostarczają opisu poszczególnych obiektów domeny, wartości ich atrybutów i opisu relacji w sposób formalny, zestandaryzowany i możliwy do przetwarzania przez komputery.

Ontologie - dostarczają interpretacji metadanych, umożliwiają ich wykorzystanie. Zapewniają jednolite i powszechne zrozumienie dziedziny dla ludzi i komputerów. Służą jako rodzaj (uzgodnionej) reprezentacji wiedzy o świecie lub jego fragmencie. Opisują elementy dziedziny,

klasy, atrybuty, relacje, akcje, zdarzenia. Są to metadane zawierające bazę wiedzy dziedzinowej, której dotyczą opisywane dane.

Przetwarzanie logiczne z wykorzystaniem reguł - umożliwia wnioskowanie wiedzy niejawnej, metadanych czy ontologii z jawnie podanych elementów. Reguły są podstawą dalszego rozwoju Sieci Semantycznej. Znajdują zastosowanie w:

- w językach opisu ontologii (alternatywa lub w powiązaniu z logiką opisową)
- wyciąganiu wniosków
- konfigurowaniu systemów
- wyrażaniu ograniczeń
- przekształcaniu danych
- reagowaniu na zdarzenia
- specyfikowaniu zachowania agentów

Języki

- formalnego opisu metadanych (anotacji) - RDF
- ontologii - OWL
- język reguł możliwe do przetworzenia przez komputer/aplikację (np. Apache Jena)

Istotne jest dostarczanie schematu specyfikacji składni (XML) i semantyki tych języków w jednolity i spójny sposób. Celem jest przetłumaczenie tych różnych języków do wspólnego języka "bazowego" (CL lub Lbase), dostarczającego jednej spójnej teorii modelu.

Narzędzia - przyjazne dla użytkownika, a niezbędne do:

Opisywania / wprowadzania metadanych (ręcznej anotacji kontekstu) czy automatycznej generacji metadanych

- tworzenia i sprawdzania poprawności ontologii
- pozyskiwania wiedzy (w postaci reguł)
- parsowania i przetwarzania języka naturalnego

Aplikacje/serwisy - wykorzystanie metadanych, ontologii, reguł, języków oraz narzędzi związanych z Siecią Semantyczną umożliwia dostarczenie skalowalnych/dostosowanych aplikacji webowych oraz Web Serwisów. Przez to sieć WWW ma być postrzegana przez ludzi jako inteligentna sieć WWW.

Podstawa infrastruktury Sieci Semantycznej - język XML i języki znacznikowe.

//Konkretne standardy opisu poniżej

XML

- standard, będący podstawą do realizacji wielu technologii internetowych
- uniwersalny, służy do tworzenia struktur danych
- elastyczny - można dodawać własne tagi
- łatwy w użyciu
- rozszerzalny - użytkownik sam definiuje tagi, ich nazwy, strukturę, atrybuty, znaczenie i uporządkowanie

- umożliwia transfer struktur za pomocą protokołów sieciowych
- wykorzystuje przestrzeń nazw (można wykorzystać te same tagi)
- zastosowanie: zbiór reguł służących do definiowania elementów (tagów), które dzielą dokument na części i tworzą w ten sposób zhierarchizowaną strukturę dokumentu
- dane prezentowane w sposób zrozumiały dla aplikacji
- definiowanie przez użytkownika elementów i sposobu ich wyświetlania

DTD (Document Type Definition)

- dokument zewnętrzny dla dokumentu XML
- Zawiera:
 - tagi zdefiniowane przez użytkownika: nazwy tagów + ich struktura
 - deklaracje definiujące jakie struktury danych będą umieszczane w dokumencie - definiuje zbiór używanych tagów, ich atrybuty, wzajemne powiązania
 - jeden, używany w wielu dokumentach xml
 - może zostać udostępniony przez sieć
- brak definiowania ograniczeń
- brak własnych typów danych wbudowanych w język

XSD (XML Schema Definition)

- alternatywa dla DTD
- określa schematy, które ograniczają i definiują zawartość XMLa
 - ograniczenia na wartości atrybutów i zawartość elementów
 - określenie jaki format danych ma być umieszczony w tagu

Przestrzeń nazw

- powiązanie pewnego zbioru tagów z unikalnym adresem URI, przez dodanie do odpowiedniego elementu przedrostka
- można używać tagów i atrybutów z różnej przestrzeni nazw, mimo tej samej nazwy

RDF (Resource Description Framework) to uniwersalny model reprezentacji (opisu) danych. Umożliwia opisywanie większości dostępnych typów danych. Opisuje strukturę zbiorów danych oraz związki pomiędzy poszczególnymi danymi.

RDF jest standardem, który pozwala na zapis danych w postaci grafu skierowanego. W grafie tym dane zawarte są w wierzchołkach a relacje pomiędzy nimi i własności tychże znajdują się w krawędziach.

RDF definiuje **zasoby**, które są zbiorem **stwierdzeń (wyrażeń)**.

Każde stwierdzenie ma postać **trójki: podmiot + orzeczenie + dopełnienie** (inaczej: **obiekt + atrybut + wartość**).

Cele twórców RDF

- prostota modelu - model prosty i łatwy do przetwarzania i manipulacji
- stworzenie formalnej semantyki i zależności
 - semantyka podstawą analizy znaczenia wyrażenia

- jednoznaczna składnia - łatwo zdefiniować stabilne zasady powiązań danych w RDF
- wykorzystanie URI do rozszerzania słownictwa - referencje jednoznaczne
- bazuje na XMLu

Składnia

- każde słowo (obiekt, atrybut, wartość) jednoznacznie zdefiniowane
- wspomniane już wyrażenia (trójki)
- struktura wyrażenia - graf skierowany (model można przedstawić w postaci graficznej - grafu)
- obiekty (wartości obiektów) reprezentowane są jako literały (integer, string....) lub inny zasób przez URI
- RDF obsługuje standardowe typy danych: integer, float, date, string
- literały mogą być **typowane**, tzn. dany łańcuch znaków powinien być interpretowany jako data, liczba czy też łańcuch znaków zgodny z określonym wzorcem

Cechy

- RDF zapewnia dodanie formalnej składni dla reprezentacji wiedzy
- Umożliwia wyrażanie relacji, jakie zachodzą między zasobami (np. zajęcia są prowadzone przez...)
- dzięki URI można łączyć różne dane z wielu oddalonych miejsc i baz danych
- nie można definiować relacji (np. każda kobieta jest człowiekiem), jakie zachodzą między danymi
- użytkownik definiuje słownictwo do opisu zasobów, brak definicji semantyki
- stanowi podstawę do reprezentowania i przetwarzania metadanych
- model można przedstawić w postaci graficznej (graf skierowany)
- składnia XML-owa umożliwia przetwarzanie syntaktyczne (składni)
- umożliwia przyrostowe reprezentowanie wiedzy, jej współdzielenie i wykorzystanie
- nie zależy od reprezentowanej dziedziny

Wykorzystanie

- metadane - informacje o zasobach Sieci Semantycznej i systemach wykorzystujących te zasoby (np. dane o pliku dźwiękowym)
- aplikacje wymagające otwartych i rozszerzalnych modeli
- model danych, które można swobodnie przetwarzać poza środowiskiem (dla którego zostały stworzone). Model pozwalający na łączenie danych z różnych źródeł i ich przetwarzanie
- zautomatyzowane przetwarzanie danych zgromadzonych w sieci semantycznej

Dane (zasoby sieciowe) opisywane są za pomocą URI, dzięki temu są one unikatowe. Odnośniki URI umożliwiają opisywanie i jednoznaczne identyfikowanie poszczególnych zasobów. Korzystanie z odnośników URI zmniejsza czytelność, dlatego częściej posługuje się etykietami do nazywania poszczególnych zasobów lub przestrzeniami nazw. Nie każdy zasób musi być

identyfikowany za pomocą odnośnika URI, istnieją zasoby, których istnienie i tożsamość jest zdeterminowana przez związki z innymi zasobami.

Zalety RDF

- umożliwia integrację danych z wielu różnych schematów,
- pozwala na całkowicie elastyczne manipulowanie strukturą poszczególnych zasobów, ponieważ własności każdego zasobu są niezależne od pozostałych zasobów,
- ułatwia wielokrotne wykorzystanie tych samych danych w każdej aplikacji potrafiącej przetwarzać dane RDF,
- pozwala na wnioskowanie nowych danych z istniejących danych na podstawie reguł wnioskowania zdefiniowanych w ontologii,
- pozwala wykorzystywać ontologie zewnętrzne w stosunku do zbioru danych,
- nie ogranicza danych do lokalnej bazy danych, ponieważ odnośniki URI definiujące zasoby mogą wskazywać na dowolne zasoby leżące poza lokalną składnicą RDF.

Wady RDF

- znacznie wolniejsze i mniej efektywne przetwarzanie danych niż w przypadku relacyjnych baz danych
- mało czytelna struktura dokumentu
- nadmiarowość modelu - w przypadku danych relacyjnych własność (definicja atrybutu) jest specyfikowana tylko raz, w nagłówku tabeli. W modelu RDF każdy zasób (odpowiadający z grubsza krotce w tabeli), posiadający pewną własność, musi deklarować posiadanie tej własności niezależnie.
- kontrowersje z URI - nie odpowiada adresowi WWW
- możliwość pojawiania się sprzecznych danych
Każda trójka może być dowolnym wyrażeniem dotyczącym dowolnego zasobu, pojawienie się konfliktowych lub sprzecznych informacji jest tylko kwestią czasu, tym bardziej, że RDF funkcjonuje w otwartym i rozproszonym środowisku Sieci. Oczywiście wnioskowanie w obecności sprzeczności jest znacząco utrudnione i może często prowadzić do mylnych wniosków. W chwili obecnej RDF nie posiada żadnych mechanizmów radzenia sobie ze sprzecznościami (choć może je wykrywać).
- RDF nie zawiera możliwości negowania wyrażień
- nie umożliwia formułowania wyrażień dotyczących ogólnych klas zasobów (np. wyrażień dotyczących wszystkich pracowników).

RDF nie jest wystarczający aby poprawnie zamodelować bardziej złożone zależności między klasami i własnościami, np. możliwość specyfikowania liczności związków (pracownik może pracować w co najwyżej dwóch zespołach) lub określania związków wymaganych (każdy pracownik musi być przypisany do co najmniej jednego projektu).

RDF definiuje **zasoby**, które są zbiorem **stwierdzeń (wyrażeń)**.

Każde stwierdzenie ma postać **trójki: podmiot + orzeczenie + dopełnienie** (inaczej: **obiekt + atrybut + wartość**).

RDFS (Resource Description Framework Schema)

- język reprezentacji wiedzy, opisu ontologii
- stanowi rozszerzenie RDF - **pojęcia** opisane są w nim opisane przez definiowanie **klas** i opisywanie ich **własności, ograniczeń oraz relacji**
- pojęcie = klasa, reprezentuje zbiór obiektów
- obiekt - instancja klasy (type)
- klasy reprezentowane są przez nazwy i przypisane im wartości (wykorzystują URI)
- za pomocą **klas, własności i relacji** (np. należenie - zawieranie), można lepiej definiować pojęcia (subclassof, domain, range)
- wykorzystuje RDF do definicji obiektów, ale dodaje do niej semantykę (klasy + ich własności, hierarchia klas + dziedziczenie, hierarchia własności)
- dokument RDFS
 - schemat
 - definiuje niewielką klasę pojęć, które należą do bardzo ograniczonej gałęzi wiedzy

Cechy

- dostarcza mechanizmy do opisywania specyficznych dziedzin
- prosty język opisu ontologii
- brak mechanizmów wnioskowania
- problem z opisem zasobów na odpowiednim poziomie szczegółowości
 - brak ograniczeń określania dziedziny i zakresu wartości
 - brak ograniczeń liczności
 - brak własności przechodnich, odwrotnych i symetrycznych

RDFS wprowadzają do grafów takie pojęcia jak klasy i podklasy, pozwalające na wspólne grupowanie danych mających cechy wspólne. Dowolna dana może znajdować się w wielu klasach.

OWL

- od 2004 roku standard modelowania ontologii (W3C)
- przeznaczony do definiowania semantyki dokumentów przez specyfikowanie wiedzy dziedzinowej
- bazuje na RDF/RDFS i DAML+OIL
 - wykorzystuje składnię RDF (oparta na XML)
 - klasy i obiekty definiowane jak w RDFS
 - dodatkowe relacje między klasami
- kompromis między siłą ekspresji języka i wydajnością mechanizmów wnioskowania
- przetwarzanie, żeby uzyskać wiedzę niejawną - wysoka złożoność obliczeniowa lub nawet nierozstrzygalność (brak odpowiedzi TAK/NIE)
- umożliwia tworzenie ontologii jako zbioru definicji pojęć, właściwości (relacji, atrybutów), instancji (obiektów, które są przykładami pojęć) i relacji
- ontologie OWL mogą być modularne - można łączyć ontologie opisane na różnym poziomie szczegółowości

OWL jest standardem pozwalającym na definiowanie klas na podstawie własności danych, a także na definiowanie logicznych charakterystyk relacji. OWL jest więc standardem formalnie zapisującym ontologię.

Wyróżnia się 3 rodzaje OWL, różniące się siłą ekspresji

- OWL Lite
 - mała siła ekspresji
 - ograniczenia - tylko licznosciowe
 - ograniczona definicja pojęć
- OWL DL (Description Logic)
 - dobrze zdefiniowana semantyka
 - dobrze określone formalne własności (złożoność, rozstrzygalność problemów)
 - znane algorytmy wnioskowania + narzędzia do wnioskowania
 - istniejące zaimplementowane systemy (o wysokiej optymalności)
 - rozstrzygalny i efektywny
 - rozszerzone definicje klas przez nakładanie kilku rodzajów ograniczeń
 - najpowszechniej wykorzystywany w Sieci Semantycznej
- OWL Full
 - zawiera pozostałe
 - wysoka siła ekspresji
 - trudny do zrozumienia i użytkowania
 - wynik może być nierozstrzygalny
 - brak wsparcia ze strony narzędzi programistycznych

23. Strategie lokalizacji użytkownika w systemach mobilnych.

http://wazniak.mimuw.edu.pl/images/0/01/Systemy_mobilne_wyklad_3.pdf

Systemy mobilne – wykład 3

Pozycjonowanie w sieciach GSM

- Identyfikator komórki
- Sektor komórki
- Sektor i czas
- Uzględnienie mapy



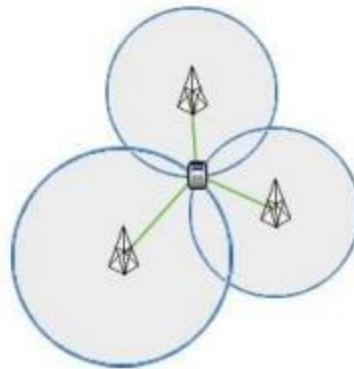
Pozycjonowanie i nawigacja użytkowników mobilnych(12)

Często stosowanym sposobem pozycjonowania użytkowników mobilnych jest wykorzystanie do tego infrastruktury sieci GSM oraz faktu posiadania przez użytkownika telefonu komórkowego. Istnieje cała rodzina metod, charakteryzujących się innymi dokładnościami pomiaru pozycji.

- z dokładnością do komórki (CELL ID). Ze względu na zróżnicowane rozmiary komórek jej dokładność waha się od kilkudziesięciu metrów do kilkudziesięciu kilometrów,
- z dokładnością do sektora komórki - bardziej precyzyjne określenie pozycji, pozwalające na jej przybliżenie do sektora (kąt rozwarty – najczęściej 135 stopni),
- z dokładnością do sektora komórki z dodatkową informacją o czasie,
- z dokładnością do sektora komórki z dodatkową informacją o czasie i powiązaniu z mapą



- Metoda pseudodległościowa
 - w oparciu o infrastrukturę
 - w oparciu o aplikację w telefonie
- Połączenie telefonu z GPS

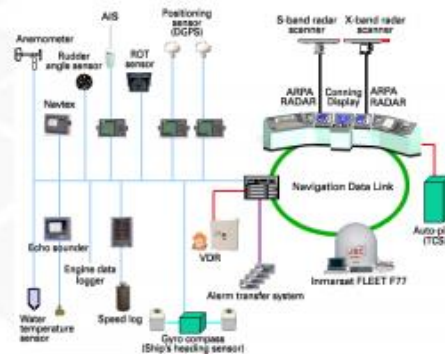


Inną znaną metodą, działającą w oparciu o infrastrukturę sieci komórkowej, jest określanie pozycji terminala na podstawie trzech lub więcej pseudodległości od stacji bazowych.

Usługi lokalizacyjne świadczone są komercyjnie i oferowane zwykłym klientom sieci GSM. Mogą być również oferowane firmom, chcącym śledzić wybraną grupę terminali. Wszyscy operatorzy są również ustawowo zobowiązani do przekazywania danych o lokalizacji terminala Policji i innym służbom.



- połączenie kilku kanałów informacji o pozycji
- zwiększenie dokładności pozycjonowania



Pozycjonowanie i nawigacja użytkowników mobilnych(18)

Zintegrowany system nawigacyjny stanowi połączenie kilku urządzeń pozycjonujących, pracujących wspólnie nad wyznaczeniem dokładnej lokalizacji użytkownika. Poza odbiornikiem GPS stosuje się w nich kompasy elektroniczne, urządzenia odometryczne, prędkościomierze i układy inercyjne.

Korzyścią jest zwiększenie dokładności pozycji poprzez pomiary z kilku źródeł i obróbkę cyfrową tych danych. Pozycja uzyskana w ten sposób jest znacznie bardziej dokładna.



- nie GPS! (tylko pseudolity)
- Wi-Fi /BT
- RF/ultradźwiękowe



•GPS nie jest systemem, którą stosuje się wewnątrz budynków z powodu braku dostępu do sygnału z satelitów Navstar. Można jednak stosować technikę GPS wykorzystując sztuczne satelity, czyli pseudolity.

•Wi-Fi/BT – pozycja użytkownika (CELL ID) jest wyznaczana na podstawie identyfikatora sieci pracującej w danym pomieszczeniu (np. SSID sieci Wi-fi)

•RF/ Ultradźwiękowe –stacja bazowa wysyła jednocześnie sygnał radiowy i ultradźwiękowy. Mobilne urządzenie użytkownika po odbieraniu sygnału RF zaczyna odliczać czas do nadejścia impulsu ultradźwiękowego. Na podstawie tego czasu i prędkości dźwięku w powietrzu wyliczana jest odległość do stacji bazowej. Na podstawie kilku odległości do różnych stacji wyznaczana jest pozycja użytkownika.

24. Sztuczna inteligencja a inteligencja obliczeniowa - podstawowe pojęcia, metody i zadania.

Definicja sztucznej inteligencji (Marvin Minsky, 1960):

– Sztuczna inteligencja jest nauką o maszynach realizujących zadania, które wymagają inteligencji gdy są wykonywane przez człowieka.

Silna i słaba sztuczna inteligencja:

- Silna SI - cecha sztucznego systemu pozwalająca mu myśleć na poziomie intelektualnym zbliżonym do człowieka a nawet go przewyższającym
- Słaba SI - zbiór metod rozwiązujących problemy które, wykonywane przez człowieka, są dla niego wymagające intelektualnie

Obliczenia inteligentne (także Metody inteligencji obliczeniowej) – część działu sztucznej inteligencji. Obliczenia inteligentne to grupa heurystycznych algorytmów, takich jak: systemy oparte na logice rozmytej oraz sztuczne sieci neuronowe i obliczenia ewolucyjne. Ponadto do metod inteligencji obliczeniowej zaliczyć można uczenie maszynowe, metody regresji i estymacji, statystykę, teorię filtrów adaptacyjnych, modelowanie Bayesowskie, logikę rozmytą, teorię zbiorów przybliżonych, algorytmy ewolucyjne, metody drażenia danych, modelowanie koneksjonistyczne, neuroinformatykę^[1]. Większość modeli obliczeniowych wyrosłych z powyższych dziedzin ma wspólną cechę, a mianowicie są to "metody uczenia się z danych".^[1]

Obliczenia inteligentne są nazywane również *obliczeniową sztuczną inteligencją* lub z ang. *Soft-computing*.

Computational Intelligence (CI)

Zajmuje się rozwiązywaniem problemów, które nie są efektywnie algorytmizowalne. Przykłady problemów niealgorytmizowalnych całkowicie lub częściowo niealgorytmizowalnych:

- rozumienie sensu zdań,
- rozwiązywanie nietypowych problemów,
- działania twórcze, decyzje intuicyjne;
- rozpoznawanie twarzy i obrazów,
- rozpoznawanie pisma ręcznego,
- rozpoznawanie mowy i sygnałów, percepcja,
- sterowanie robotem, nieliniowymi układami,
- diagnostyka medyczna, planowanie terapii.

Wiele tego typu problemów nie ma natury kombinatorycznej podobnie jak problemy optymalizacyjne.

CI: percepcja i sterowanie: zachowania sensomotoryczne – sieci neuronowe i uczenie maszynowe.

AI: wyższe czynności poznawcze: logika, język, rozumowanie, rozwiązywanie problemów.

http://www.is.umk.pl/~duch/Wyklady/NN_plan.html

Zadania:

Wnioskowanie

- Podstawa: logika (w różnej formie)
- Zastosowania:
 - Systemy eksperckie
 - Systemy automatycznego dowodzenia twierdzeń
 - Gry komputerowe

Przeszukiwanie

- Algorytmy ewolucyjne i genetyczne
- Algorytmy znajdujące optymalne sekwencje, w tym algorytmy planowania
- Algorytmy wyboru ruchu w grze: MIN-MAX, α - β

Uczenie maszynowe

- Uczenie się wiedzy deklaratywnej
- Uczenie proceduralne (ze wzmocnieniem)
- Sieci neuronowe

Systemy rozmyte

- W stronę wnioskowania tak jak to robi człowiek
- Rozmyte pojęcia
- Rozmyte reguły wnioskowania

dziedziny pokrewne

- Inne gałęzie
 - Inteligencja roju
 - Sztuczne życie
 - Kognitywistyka

Metody sztucznej inteligencji są przedstawiane w języku dziedzin pokrewnych: informatyki, matematyki, ekonomii, badań operacyjnych, teorii podejmowania decyzji itd.

- Metody sztucznej inteligencji przechodzą do zastosowań jako narzędzia z repertuaru informatyki. Występują w analizie danych, wyszukiwarkach, robotyce przemysłowej, logistyce, rozpoznawaniu mowy, grach komputerowych, bankowych systemach inf. itd.
- Nowe pomysły są przedstawiane w sposób ścisły
- Nowe metody są niekiedy prezentowane pod szyldem innych, historycznie starszych dyscyplin

Inteligencja sztuczna

- Obliczenia twarde
(ang. hard computing)
- Logika binarna
 - Prawda i fałsz
- Wiedza kodowana ręcznie
- Metody klasyczne
- Siłowe przeszukiwanie

Zastosowania Metod Inteligencji Obliczeniowej

Inteligencja obliczeniowa

- Obliczenia miękkie
(ang. soft computing)
- Logika rozmyta
 - Prawdopodobieństwa
 - Współczynniki pewności itp.
- Odkrywanie wiedzy z danych
- Symulacja procesów naturalnych
- Symulacja inteligencji ludzkiej

Inteligencja obliczeniowa

- Obliczenia miękkie
(ang. soft computing)
- Logika rozmyta
 - Prawdopodobieństwa
 - Współczynniki pewności itp.
- Odkrywanie wiedzy z danych
- Symulacja procesów naturalnych
- Symulacja inteligencji ludzkiej

Zastosowania Metod Inteligencji Obliczeniowej

Uczenie maszynowe

- Obliczenia miękkie
(ang. soft computing)
- Logika rozmyta
 - Prawdopodobieństwa
 - Współczynniki pewności itp.
- Odkrywanie wiedzy z danych
- Metody bazujące na statystyce
- Maksymalizacja sztucznych współczynników jakości



Podstawowe pojęcia (3)

- **Inteligencja obliczeniowa (CI- computational intelligence)**-zbiór metod, które zajmują się rozwiązywanie obliczeniowo problemów, które nie są efektywnie algorytmizowane
- W inteligencji obliczeniowej wykorzystywane są metody matematyczne oraz inspiracje: biologiczne, biocybernetyczne, psychologiczne, statystyczne, logiczne, informatyczne i inne

Na podstawie: W. Duch: *Dokąd zmierza inteligencja obliczeniowa?*



Podstawowe pojęcia (4)

- **Sztuczna inteligencja (AI- artifical intelligence)**-część inteligencji obliczeniowej, zajmująca się pamięcią semantyczną i wyższymi czynnościami poznawczymi, związanymi ze zdolnością do rozumowania, planowania, tworzenia teorii i posługiwania się językiem



Różnice między AI a CI?

- AI oparta na przetwarzaniu symbolicznym, w CI nie używa się symboli
- W AI mamy narzuconą strukturę i wypełniamy ją danymi, w CI mamy odwrotnie- struktura pojawia się na końcu.
- W CI struktury mogą wyewoluować, ale nie są narzucone od samego początku.
- AI jest częścią CI bez narzuconych struktur.
- CI nazywamy często jako *Soft Computing*- nie mamy gwarancji znalezienia rozwiązania



Cele AI i CI

- AI: maszyna nieodróżnialna od człowieka przy zdalnej konwersacji, która przejdzie test Turinga. Wymaga nie tylko zdolności lingwistycznych, ale i budowania modeli umysłowych, szerokiej wiedzy o świecie, zrozumienia stanów emocjonalnych
- CI: sztuczny robot, który przetrwa we wrogim środowisku. Wymaga percepcji, kontroli, pamięci skojarzeniowej, planowania, antycypacji ...

25. Wielomodalna interakcja człowiek-komputer.

[http://ki.uni.lodz.pl/cbolek/ick/ick_03-mat\(4\).pdf](http://ki.uni.lodz.pl/cbolek/ick/ick_03-mat(4).pdf)

To ponizej malo ma wspolnego, ten pdf jest spoko:

https://multimed.org/student/tm/tm14_multimodalne.pdf

Interakcja człowiek-komputer - tryb pracy polegający na naprzemiennej wymianie informacji między użytkownikiem a komputerem, w którym użytkownik wydaje komputerowi polecenia, których skutki może na bieżąco obserwować i podejmować uzależnione od nich decyzje.

Systemy interakcyjne - klasa systemów, których działanie wymaga znacznego stopnia interakcji z użytkownikiem. Podstawową cechą systemów interakcyjnych jest wzajemna wymiana informacji pomiędzy użytkownikiem a systemem komputerowym.

Style interakcji - alternatywna strategia projektowania i realizacji interfejsu użytkownika

- klawiszowo-modalny
- bezpośredniej manipulacji
- multimodalne
- niebazujące na komendach
- wirtualna rzeczywistość

Styl klawiszowo-modalny

- interfejs obsługiwany przez pewną liczbę klawiszy (przycisków)
 - przyciski pełnią różną funkcję w zależności od kontekstu
 - interakcja oparta na menu
 - system generuje zestaw opcji wywołujących inne akcje lub generujące nowe opcje
 - najbardziej popularny styl interakcji
 - różne typy (pop-up, pull-down)
 - powstały przewodniki stylów dla niektórych producentów
- pytania i odpowiedzi
 - pytania w formie tekstowej
 - oczekiwana odpowiedź wpisana na klawiaturze
 - pytania pojedynczo, mogą zależeć od odpowiedzi
 - bankomaty, katalogi biblioteczne
- klawisze funkcyjne
 - specjalne klawisze
 - informacja zwrotna wyświetlana na osobnym urządzeniu
 - przyciski, klawisze funkcyjne, klawiatura numeryczna
 - kody kreskowe, karty magnetyczne
 - wymagane pokazywanie stanu systemu (zapytania, komunikaty)
 - automaty (batony, bilety), pralka, pilot TV,
- komendy prezentowane głosowo (interakcja głosowa)
 - nagrany wcześniej głos - automatyczna sekretarka

- przeładowanie pamięci użytkownika - trudno mu podjąć decyzję o wyborze opcji

Styl bezpośredniej manipulacji

- interfejs wyświetla wiele obiektów naraz, użytkownik wykonuje na nich odpowiednie akcje
- GUI - bezpośrednia manipulacja graficzna
 - standard interakcji człowiek-komputer
 - obiekty 2D/3D (głównie w grach)
- wypełnianie formularzy, uzupełnianie pól tekstowych
 - efektywna realizacja złożonych zadań, np. wyszukiwania i wprowadzania danych

Styl lingwistyczny

- wprowadzanie danych za pomocą klawiatury, czytnika pisma, mikrofonu lub innego wejścia
- linia komend
 - szybkie wprowadzanie danych
 - długi czas uczenia się
- tekst w języku naturalnym
 - ograniczony zbiór poleceń - użytkownik musi wiedzieć, co rozumie komputer

Styl multimodalny

- nie bazuje na komendach
- ciągle monitorowanie użytkownika w celu uchwycenia intencji
- jednoczesna analiza wielu źródeł, aby dobrze zinterpretować zachowanie
- głos, ruchy oka, żrenicy, gestów, ruchów ciała
- zastosowanie interfejsów znak + ograniczenie - wymagają operatorów i operandów

Wirtualna rzeczywistość

- środek wizualizacji, manipulacji, interakcji człowieka z komputerem i złożonymi danymi
- hełm, rękawice, kombinezon, urządzenia generujące dźwięk przestrzenny
- intensywność - zwrócenie uwagi na wybranych obiektach
- interaktywność - współdziałanie użytkownika z systemem przez interfejs
- immersja - absorbowanie uwagi przez trójwymiarowe bodźce
- ilustracyjność - informacje podawane w sposób przejrzysty, opisowy, jasny
- intuicyjność - informacje łatwe do interpretacji

Wybór stylu interakcji

- informacje o użytkownikach, zadaniach jakie będą realizowali
- możliwe do zastosowania technologie
- metoda zawężania wyboru - do 1 lub 2 stylów
- pomocne mogą być zasady projektowania zorientowanego na użytkownika - UCD

26. Zarządzanie ryzykiem w projektach informatycznych.

Ryzyko - prawdopodobieństwo wystąpienia niepożądanego zdarzenia oraz jego wpływ na cele projektu (Ryzyko prowadzi do powstania problemu).

Ryzyko to prawdopodobieństwo, że w danym punkcie cyklu życia oprogramowania, jego zaplanowane cele nie zostaną osiągnięte w ramach dostępnych zasobów.

Zarządzanie ryzykiem to unikanie lub neutralizacja skutków niepomyślnych dla projektu wydarzeń.

Zarządzanie ryzykiem to systematyczny proces identyfikowania, analizowania i reagowania na ryzyka w projekcie. Obejmuje maksymalizację prawdopodobieństwa wystąpienia i wpływu pozytywnych zdarzeń na cele projektu oraz minimalizację prawdopodobieństwa wystąpienia i wpływu negatywnych zdarzeń.

W czasie procesu zarządzania ryzyka są oceniane i systematycznie obniżane do akceptowanego poziomu.

Efektywny menedżer musi identyfikować i osłabiać ryzyka w czasie całego cyklu życia systemu.

Nie można wyeliminować wszystkich ryzyk, trzeba skupiać się na najbardziej krytycznych ryzykach dla powodzenia przedsięwzięcia do momentu aż osiągną akceptowalny poziom.

Ryzyka

- przekroczenie budżetu
- zmiana założeń
- przekroczenie ram czasowych
- niespełnianie potrzeb użytkownika

Na ryzyko wpływają głównie:

- wielkość projektu
- doświadczenie technologiczne
- uniwersalność systemu
- złożoność systemu

Cykl zarządzania ryzykiem

- **Identyfikacja** - poszukiwanie i określanie ryzyk
- **Analiza** - przetwarzanie danych o ryzyku w informacje niezbędne do podjęcia decyzji
- **Planowanie** - informacje o ryzyku są przekształcane w decyzje i działania, wdrożenie działań
- **Śledzenie** - monitorowanie wskaźników ryzyka
- **Kontrola** - korygowanie odchyłeń od zaplanowanych działań postępowania z ryzykiem
- **Komunikacja** - dostarczanie informacji i zbieranie informacji zwrotnych na temat działań postępowania z ryzykiem, bieżących i powstających ryzyk



Techniki identyfikacji ryzyka

- przegląd dokumentacji projektu
- gromadzenie informacji
 - burza mózgów
 - metoda delficka - przewidywanie na podstawie wiedzy, doświadczenia i opinii ekspertów z danej dziedziny
 - Wywiady
 - SWOT (Strengths, Weaknesses, Opportunities, Threats) (mocne strony, słabe strony, szanse, zagrożenia)
 - mocne i słabe strony - czynniki wewnętrzne
 - szanse i zagrożenia - czynniki zewnętrzne
- listy kontrolne - lista z możliwymi ryzykami
- analiza założeń - taksonomia?
- techniki oparte na diagramach
 - Przyczynowo-skutkowe
 - przepływu dla systemu i procesów
 - wpływu i oddziaływania

Taksonomia ryzyka projektu informatycznego

- identyfikuje i objaśnia techniczne i menedżerskie niepewności i problemy
- 3 klasy
 - **inżynieria produktu**- techniczne aspekty pracy do wykonania

- **środowisko rozwoju**- metody, procedury i narzędzia użyte do wytworzenia produktu
- **ograniczenia programu**- czynniki kontraktowe, organizacyjne i operacyjne, w ramach których opracowywane jest oprogramowanie, lecz które zazwyczaj jest poza bezpośrednią kontrolą lokalnego zarządu
- klasy dzielone na elementy i atrybuty

Diagram procesu decyzyjnego (Process Decision Program Chart - PDPC)

- graficzna reprezentacja alternatywnych działań dla realizacji planu w warunkach ryzyka
- używane przy tworzeniu planów do identyfikacji potencjalnych zagrożeń
- po zidentyfikowaniu ryzyk używany do określenia i wyboru zbioru możliwych środków zaradczych
- używany do planowania sposobów unikania i eliminacji zidentyfikowanych ryzyk
- Plan->możliwy problem->możliwy środek zaradczy

Postępowanie z ryzykiem

- **Unikanie ryzyka**- podejmowanie alternatywnych działań, nie tych związanych z ryzykiem
- **Obniżanie ryzyka** - działania obniżające, ale nie eliminujące zagrożenia
- **Plany awaryjne** - nie obniżają prawdopodobieństwa wystąpienia ryzyka, dodatkowe plany działania jak wystąpi ryzyko - zmniejszenie kosztów i zakłóceń

Miara ryzyka - porównywanie ryzyk

Efektywność redukcji ryzyka - porównanie środków zaradczych

Analiza i ocena ryzyka

- oszacowanie wpływu ryzyka na przebieg projektu
- prawdopodobieństwo, że ryzyko wystąpi

Miara ryzyka = prawdopodobieństwo wystąpienia x strata spowodowana ryzykiem

ang. RI – Risk Impact lub RE – Risk Exposure

$$Ryzyko = \sum_i^n \text{prawdop}(r_i) \cdot \text{strata}(r_i)$$

Metoda punktowa szacowania ryzyka

- każdy element określony za pomocą pojedynczej wartości liczbowej
- podział zadań na kategorie ryzyka
- wagi dla ryzyk
- akcje zapobiegające ryzyku

efektywność redukcji ryzyka = oszczędność z redukcji / koszt redukcji

ang. RRL – Risk Reduction Leverage

$$RRL = \sum_i^n \frac{strata_{przed}(r_i) - strata_{po}(r_i)}{koszt_{redukcji}(r_i)}$$

Najlepiej się posłużyć przykładem (światowym standardem) metody kontroli realizacji projektu:

Earned Value Method (Metoda wartości uzyskanej):

- Głównym celem EVM jest szacowanie całkowitych kosztów projektu na podstawie rozbieżności pomiędzy monitorowanymi aktualnymi kosztami i postępami prac projektowych a ustalonym harmonogramem i planem uruchamiania środków finansowych w projekcie.
- Opiera się na obliczaniu kilku podstawowych wskaźników oraz śledzeniu ich trendów i odchył.
- Stosowanie metodyki Earned Value daje dobre rezultaty dla projektów trwających ponad rok.
- Zgodnie z przyjętymi kryteriami pierwsze szacowanie należy wykonać na etapie ok. 15% zaawansowania projektu, a kolejne najlepiej co miesiąc.
- metoda ta pozwala nie tylko na obiektywną ocenę postępów projektu, ale również na prognozowanie jego przyszłych parametrów,
- sposób mierzenia postępu realizacji prac projektu z uwzględnieniem wykonania zadań i kamieni milowych,
- obliczanie stopnia zaawansowania zadania na podstawie stosunku faktycznego nakładu pracy do nakładu planowanego.

Podstawą stosowania metody EarnedValue jest wyznaczenie Wartości Uzyskanej (BCWP - BudgetedCostof WorkPerformed, WU), obliczanej ze wzoru:

WU = budżet zadania * % wykonania zadania

Znacznie trudniejszą rzeczą jest określenie na daną chwilę procentu wykonania zadania, szczególnie dla zadań długotrwałych. Istnieje wiele technik, które umożliwiają oszacowanie tej wartości:

- Metoda 0/100

Najprostsza z metod. Polega na przydzielaniu wartości uzyskanej w wysokości 100% planowanych kosztów dopiero po całkowitym wykonaniu zadania. Jest ona dobra dla małych krótkotrwałych zadań.

- Metoda 50/50

Jest podobna do poprzedniej, jednak po rozpoczęciu zadania uzyskuje ono 50% planowanego budżetu. Kolejne 50% przydzielane jest dopiero po ukończeniu zadania. Gdy mamy dużo takich zadań, statystyczny rozkład ich rozpoczęć i zakończeń powoduje, że krzywa wartości uzyskanych dobrze odpowiada rzeczywistości.

- Itd..

Zalety EVM:

1. Kompleksowa kontrola realizacji projektu w aspekcie rzeczowym, czasowym i kosztowym
2. Obiektywna miara postępu projektu, wyrażona w jednej jednostce niezależnie od różnych jednostek obmiaru poszczególnych prac
3. Identyfikacja rodzaju i wielkości odchyłeń budżetu
4. Analiza trendu wydajności realizacji projektu
5. System wczesnego ostrzegania (wczesna prognoza oszacowania kosztu końcowego)
6. Możliwość analizy na różnych poziomach struktury podziału prac (WBS) i struktury organizacyjnej (OBS)
7. Prostota - kilka, intuicyjnie zrozumiałych wskaźników
8. Standard wymiany informacji o projekcie na poziomie wyższego kierownictwa
9. Dobrze udokumentowany, powszechnie znany i stosowany standard kontroli realizacji projektów

27. Zarządzanie zespołami ludzkimi w projektach informatycznych.

Większość profesjonalnego oprogramowania tworzona jest przez zespoły składające się z dwustu do kilkuset osób. Nie ma możliwości, żeby wszyscy członkowie tak dużego zespołu pracowali razem nad jednym problemem. Dlatego zespoły dzieli się na kilka mniejszych grup, z których każda pracuje nad inną częścią danego systemu. Przyjmuje się zasadę, że grupy powinny liczyć do 8 osób. Utworzenie wydajnie pracującej grupy to zadanie menedżera. W grupie powinna panować równowaga umiejętności technicznych, doświadczenia i osobowości.

KIEROWNIK PROJEKTU

- cel: dostarczenie produktu w wymaganym czasie, w ramach określonego budżetu, posiadającego odpowiednią jakość
- Funkcje
 - Planowanie
 - Organizowanie
 - Motywowanie
 - Kontrolowanie
- odpowiedzialność za planowanie i prognozowanie
 - odpowiedzialność interpersonalna (prowadzenie zespołu, komunikacja z klientem, zarządem)
 - odpowiedzialność za stan informacji (monitorowanie wydajności, postępu prac, informowanie o zadaniach, stanie projektu)
 - odpowiedzialność decyzyjna (zarządzanie zasobami, negocjacje kontraktów)
- zaspokajają potrzeby jednostki, zespołu, zadania
- rozwiązuje konflikty (kompromis, łagodzenie, narzucenie/przymus, unikanie)

MOTYWOWANIE

Zadaniem kierownika jest motywowanie zespołu jako całości, jak i każdej osoby z osobna. **Motywacja zespołu** ma źródło w osobistym zaangażowaniu kierownika, w sposobie przydziału i podziału pracy, jasnej wizji celu i sposobu jego osiągnięcia. Kierownik daje przykład przez własne zaangażowanie i zachowanie.

Motywację osobistą osiąga się przez stosunek między tymi osobami i "niepisaną umowę", czego dana osoba i kierownik od siebie oczekują.

Kluczowym elementem motywacji jest projektowanie pracy poszczególnych osób. Składa się na nią: odpowiednia ilość wyzwań, różnorodność zadań, perspektywa osiągnięcia celu (wyniku).

Ludzi motywuje się przez spełnienie ich potrzeb

- Potrzeby **samorealizacji**
- Potrzeby **szacunku**
- Potrzeby **społeczne**
- Potrzeby bezpieczeństwa
- Potrzeby fizjologiczne

3 ostatnie są najważniejsze z punktu widzenia kierownika.



ERGONOMIA PRACY

- mniejsze pomieszczenia (np. zespół w 1 pomieszczeniu)
- personalizacja stanowisk pracy
- pokój zebrań formalnych i nieformalnych
- praca na nowoczesnym sprzęcie
- atmosfera, równomierny rozkład pracy, brak przenoszenia odpowiedzialności.
- komfort psychiczny

ETAPY ROZWOJU ZESPOŁU

- Formowanie
- Burza
 - opór przed zadaniami, uzgadnianie
 - lider: rozwiązuje nieporozumienia
- Normowanie
 - proces burzy daje uzgodnione podejście do podejmowania decyzji, członkowie zespołu znają i akceptują role, zadania, ...
 - lider: rozwiązuje ewentualne konflikty, bardziej obserwuje niż interweniuje
- Wykonywanie
 - zespół pracuje najbardziej efektywnie
 - lider: zarządza wykonywaniem, doradza, zapewnia szkolenia, informacje zwrotne
- Przechodzenie
 - koniec projektu, pracownicy przechodzą do innych zadań
 - lider: na zebraniach podsumowuje pracę, motywuje do dalszej

AGILE

W zarządzaniu zespołami obecnie odchodzi się od podejścia opartego na "władzy" i ścisłej kontroli. Zamiast tego wspiera się samowystarczalność i samodzielność zespołów, stwarza się

im warunki do wykonania powierzonej im pracy, zachęca do rozwoju (samodzielnego lub przez szkolenia). Coraz większą popularność zyskują metodyki zwinne (agile), nastawione na ludzi i ich komunikację. "Ludzie najważniejszym czynnikiem decydującym o sukcesie lub porażce projektu".

XP

- główny cel: sprawny rozwój oprogramowania:
 - niższe koszty,
 - mniejsza ilość błędów,
 - większa produktywność
- zmiany szansą na rozwój i doskonalenie
- nacisk na komunikację ustną
- praca stałym rytmem, bez nadgodzin
- Wartości
 - naciska na komunikację
 - informacje zwrotne od klientów, użytkowników
- Zasady:
 - ludzie tworzą oprogramowanie - praktyki są dla nich
 - ciągłe doskonalenie
 - jakość podstawą kontrolowania projektu
 - małe kroki
- Praktyki
 - niewielki, samowystarczalny zespół (10 osób)
 - praca w 1 pomieszczeniu
 - programowanie w parach
 - cykl tygodniowy, kwartalny
 - Scenariusze
 - programowanie poprzedzone testami - TDD
 - ciągła integracja - CI
 - Refaktoryzacja
 - projektowanie przyrostowe
 - kompilacje 10-minutowe
 - wspólny kod
 - klient częścią zespołu

SCRUM

- środowisko do organizacji i zarządzania pracą
- szkielet prowadzenia projektu, który można łączyć z innymi podejściami, technikami
- skupia się na usprawnianiu zarządzania projektem, procesu dostarczania i rozwoju oprogramowania
- 3 filary
 - przejrzystość -- jednoznaczny odbiór, wszystko udostępnione dla tego kto tego potrzebuje
 - inspekcja -- częsta kontrola do wykrywania nieprawidłowości

- adaptacja -- ograniczenie skutków nieprawidłowości, uniknięcie dalszych problemów
- User stories - historyjki użytkownika? - lista wymagań, krótkie, na małej kartce + kryteria akceptacji
- Product backlog - Rejestr produktu? - user stories + priorytety
- **Sprint**
 - iteracja - od tygodnia do miesiąca (wszystkie tyle samo)
 - planowanie - wybór najważniejszych elementów z *product backloga* i stworzenie *sprint backloga*- rejestr sprintu?
 - zespół sam wybiera kolejność zadań do realizacji
 - codzienny stand-up do 15 minut
 - wykresy spalania, wypalania
 - na koniec przegląd sprintu (z interesariuszami) + retroperspektywa (analiza zespołu)
- **Zespół Scrumowy**
 - 5-9 osób
 - samowystarczalny, interdyscyplinarny
 - Właściciel Produktu (ang. Product owner)
 - Scrum Master
 - Zespół developerski

Scrum vs agile:

<https://stackoverflow.com/questions/11469358/what-is-the-difference-between-scrum-and-agile-development>

CRYSTAL

- rodzina metodyk, zbiór przykładów
- bazuje na podziale projektów pod względem krytyczności (błąd powoduje utratę czegoś) i liczby osób (nawiązuje do kolorów i twardości kryształów)
- programowanie jako gra zespołowa pomysłowości i komunikacji, gra skończona
- 4 punkty krytyczne
 - nacisk na komunikację
 - obecność ekspertów w miejscu pracy zespołu
 - przyrostowe tworzenie oprogramowania (co miesiąc)
 - zautomatyzowane testy
- zespół może wybrać dowolne praktyki jakie zna
 - wymagane:
 - częste dostarczanie
 - komunikacja osmotyczna
 - analiza + poprawianie metodyki

28. Zastosowanie inteligencji biznesowej do weryfikacji hipotez i zwiększania KPI (ang. Key Performance Indicators).

KPI (key performance indicators), czyli kluczowe wskaźniki efektywności to mierzalne wartości, które pokazują, jak skutecznie osoby zaangażowane w biznesie osiągają najważniejsze dla nich cele biznesowe. Wykorzystywane są w wielu różnych dziedzinach i pozwalają ocenić skuteczność działań biznesowych. Pozwalają ograniczyć dużą ilość informacji do najbardziej potrzebnych danych, które dają najlepszy obraz realizacji wyznaczonych celów. Wskaźniki KPI są charakterystyczne dla danej firmy, konkretnego stanowiska i będą się różnić w zależności od charakterystyki przedsiębiorstwa, branży, wielkości firmy bądź strategii działania.

[Przykłady KPI w zależności od obszaru działalności / modelu biznesowego](#)

[Finanse](#)

- EBITDA
- ROE (rentowność kapitałów własnych)
- Przepływy pieniężne z działalności operacyjnej
- Przychody ze sprzedaży
- Zadłużenie netto / EBITDA, gdzie Zadłużenie netto = zobowiązania – środki pieniężne

[SaaS](#)

- MRR (Monthly Recurring Revenue)
- CAC (cost of customer acquisition)
- LTV lifetime value
- RCS (Recurring Costs of Service)
- Customer Churn Rate

[Sprzedaż](#)

- Sprzedaż netto
- Średnia wartość transakcji
- Wartość utraconych szans sprzedaży
- Wartość sfinalizowanych transakcji w stosunku do ilości zamówień
- Średni przychód przypadający na jednego handlowcę

Marketing

- Liczba nowych leadów
- Liczba subskrypcji newslettera
- Koszt pozyskania leadów
- Liczba zapytań ofertowych ze strony internetowej
- Zaangażowanie na kanałach marki w mediach społecznościowych

Obsługa klienta

- Liczba wykonanych telefonów
- Średnia ocena jakości obsługi
- Średnia liczba rozmów obsługiwanych przez jednego pracownika
- Liczba zgłoszonych reklamacji
- Średni czas rozmowy i rozwiązania problemu

E-commerce

- Zwiększenie ruchu na stronie
- Ruch na stronie wygenerowany przez poszczególne kanały
- Współczynnik konwersji
- Liczba sesji z dodaniem produktów do koszyka
- Czas spędzony na stronie

Business Intelligence applications are one way of monitoring and assessing the level of Key Performance Indicators being met. Business Intelligence systems compile data relating to profit, productivity, customer return, and marketing trends. With this information readily available managers and executives can review the data and put in place indicators as needed, then using the software on a regular basis they are able to assess the success rate of the plans they initiate.