POLITECHNIKA WROCŁAWSKA
WYDZIAŁ ELEKTRONIKI

KIERUNEK: INFORMATYKA
SPECJALNOŚĆ: INŻYNIERIA SYSTEMÓW INFORMATYCZNYCH

PRACA DYPLOMOWA INŻYNIERSKA

Szablon pracy dyplomowej
inżynierskiej/magisterskiej, wersja 0.6

Engineering/master thesis template, version 0.6

AUTOR:
Imię Nazwisko

PROWADZĄCY PRACĘ:
tytuł, Imię Nazwisko, Jednostka

WROCŁAW, 2021

Opracował: Tomasz Kubik <tomasz.kubik@pwr.edu.pl>
Data: maj 2021

Streszczenie

Streszczenie w jzyku polskim powininno zmieci si na poowie strony (drug poow powinien zaj abstract w jzyku angielskim).

Lorem ipsum dolor sit amet eleifend et, congue arcu. Morbi tellus sit amet, massa. Vivamus est id risus. Sed sit amet, libero. Aenean ac ipsum. Mauris vel lectus.

Nam id nulla a adipiscing tortor, dictum ut, lobortis urna. Donec non dui. Cras tempus orci ipsum, molestie quis, lacinia varius nunc, rhoncus purus, consectetuer congue risus.

Słowa kluczowe: raz, dwa, trzy, cztery

Abstract

Streszczenie in Polish should fit on the half of the page (the other half should be covered by the abstract in English).

Lorem ipsum dolor sit amet eleifend et, congue arcu. Morbi tellus sit amet, massa. Vivamus est id risus. Sed sit amet, libero. Aenean ac ipsum. Mauris vel lectus.

Nam id nulla a adipiscing tortor, dictum ut, lobortis urna. Donec non dui. Cras tempus orci ipsum, molestie quis, lacinia varius nunc, rhoncus purus, consectetuer congue risus.

Keywords: one, two, three, four

Spis treści

Spis rysunków

Spis tabel

Spis listingów

Skrty

-

OGC (ang. *Open Geospatial Consortium*)
XML (ang. *eXtensible Markup Language*)
SOAP (ang. *Simple Object Access Protocol*)
WSDL (ang. *Web Services Description Language*)
UDDI (ang. *Universal Description Discovery and Integration*)
GIS (ang. *Geographical Information System*)
SDI (ang. *Spatial Data Infrastructure*)
ISO (ang. *International Standards Organization*)
WMS (ang. *Web Map Service*)
WFS (ang. *Web Feature Service*)
WPS (ang. *Web Processing Service*)
GML (ang. *Geography Markup Language*)
SRG (ang. *Seeded Region Growing*)
SOA (ang. *Service Oriented Architecture*)
IT (ang. *Information Technology*)

Rozdział 1

WstÄ™p

1.1. Geneza pracy

UsĹ‚ugi sieciowe, zarÄ…wno te dostÄ™pne publicznie jak i te realizowane dla celÄ…w prywatnych, peĹ‚niÄ… kluczowÄ… rolÄ™ w kontekĹ›cie funkcjonowania wspÄ…Ĺ‚czesnej sieci internetowej. Zapewne nikt z nas, nie jest w stanie wyobraziÄ‡ sobie ksztaĹ‚tu obecnego Internetu bez takich rozwiÄ…zaĹ„ sieciowych jak obsĹ‚uga poczty elektronicznej, realizacja transferu plikÄ…w, czy teĹ¼ przede wszystkim dostÄ™p do aplikacji oraz witryn internetowych. SzczegÄ…lnie w obrÄ™bie ostatniej spoĹ›rÄ…d wymienionych usĹ‚ug, na przestrzeni ostatnich lat zauwaĹ¼yÄ‡ moĹ¼na bardzo duĹ¼Ä… liczbÄ™ zmian dotyczÄ…cych sposobu ich definiowania oraz realizacji. Powodem pojawiania siÄ™ tych zmian, jest niewÄ…tpliwie koniecznoĹ›Ä‡ zachowania bÄ…dĹş teĹ¼ zwiÄ™kszenia poziomÄ…w wydajnoĹ›ci, niezawodnoĹ›ci oraz bezpieczeĹ„stwa oferowanych rozwiÄ…zaĹ„ uwzglÄ™dniajÄ…c coraz to wiÄ™kszy ruch sieciowy, generowany przez nieustannie zwiÄ™kszajÄ…cÄ… siÄ™ liczbÄ™ uĹ¼ytkownikÄ…w Internetu. Ponadto, od nowoczesnego systemu internetowego, wymaga siÄ™ coraz to wiÄ™kszego poziomu skalowalnoĹ›ci, a takĹ¼e pĹ‚ynnoĹ›ci dziaĹ‚ania.

Poparciem niniejszych sĹ‚Ä…w, moĹ¼e byÄ‡ treĹ›Ä‡ wydawanego w kilkuletnich odstÄ™pach czasu raportu firmy Cisco, dotyczÄ…cego przewidywaĹ„ oraz trendÄ…w sieciowych (tj. Cisco Annual Internet Report). Zgodnie z przedstawionymi w przytoczonym raporcie informacjami, a takĹ¼e porÄ…wnujÄ…c informacje te, z faktycznymi wartoĹ›ciami wskaĹşnikÄ…w dotyczÄ…cych ruchu w internecie, zaobserwowaÄ‡ moĹ¼emy niemalĹ¼e trzykrotny wzrost globalnego ruchu sieciowego na przestrzeni ostatnich piÄ™ciu lat. Ponadto, liczba klienckich urzÄ…dzeĹ„ sieciowych, wykorzystywanych w celu uzyskania dostÄ™pu do usĹ‚ug udostÄ™pnianych w Internecie, na przestrzeni analogicznego przedziaĹ‚u czasowego, zwiÄ™kszyĹ‚a siÄ™ z wartoĹ›ci 2,4 urzÄ…dzenia na osobÄ™, do poziomu niemalĹ¼e czterech hostÄ…w sieciowych przypadajÄ…cych na pojedynczego reprezentanta globalnej populacji.

NaleĹ¼y takĹ¼e zwrÄ…ciÄ‡ uwagÄ™, jakiego typu ruch sieciowy peĹ‚ni dominujÄ…cÄ… rolÄ™ w kontekĹ›cie dzisiejszego Internetu. Ponad 80% globalnego konsumenckiego ruchu internetowego stanowiÄ… dane dotyczÄ…ce usĹ‚ug wideo, okoĹ‚o dziesiÄ™ciu procent Ĺ›wiatowego ruchu obejmujÄ… pozostaĹ‚e treĹ›ci udostÄ™pniane w ramach aplikacji oraz witryn internetowych, a pozostaĹ‚e 10% to ruch generowany m.in. przez usĹ‚ugi transferu plikÄ…w, poczty elektronicznej, czy teĹ¼ gier online. Na podstawie tych informacji, zauwaĹ¼yÄ‡ moĹ¼na, Ĺ¼e ponad 90% caĹ‚oĹ›ci danych, przesyĹ‚anych w ramach globalnej sieci, musi byÄ‡ przetwarzanych przez aplikacje internetowe, bÄ…dĹş usĹ‚ugi sieciowe z nimi powiÄ…zane. Dlatego teĹ¼, zaawansowane witryny internetowe komunikujÄ…ce siÄ™ z usĹ‚ugami sieciowymi, zwane dziĹ› systemami internetowymi, tworzone sÄ… z wykorzysta-

niem coraz to bardziej udoskonalonych modeli architektonicznych, pozwalajÄ…cych na coraz to Ĺatwiejszą budowÄ™ i rozwÄłj rozwiÄ…zaĹ„ przystosowanych do potrzeb aktualnego ruchu sieciowego [**?**].

Jednym z pierwszych, a takĹĽe najbardziej podstawowych podejĹ›Ä‡ do projektowania i implementacji systemÄłw internetowych byĹ‚o wprowadzenie modelu architektury definiujÄ…cego aplikacje monolityczne. W modelu tym, uĹĽytkownik aplikacji, wykorzystujÄ…c oprogramowanie klienckie, ktÄłrym w tym przypadku jest przeglÄ…darka internetowa, wysyĹ‚aĹ‚ ĹĽÄ…danie uzyskania zasobu definiujÄ…c odpowiedni adres url (*(ang. Uniform Resource Locator)*). Ĺ»Ä…danie to, odwoĹ‚ywaĹ‚o siÄ™ bezpoĹ›rednio do fizycznego zasobu zlokalizowanego na serwerze, ktÄłry przed dostarczeniem do klienta byĹ‚ przetwarzany przez serwer w celu uzupeĹ‚nienia go danymi uzyskanymi z zewnÄ™trznych ĹşrÄłdeĹ‚ - m.in. z systemu bazodanowego. Odpowiednio przygotowana statyczna zawartoĹ›Ä‡ odpowiedzi serwera, przybierajÄ…ca postaÄ‡ pliku HTML (*ang. HyperText Markup Language*) byĹ‚a nastÄ™pnie przesyĹ‚ana bezpoĹ›rednio do przeglÄ…darki internetowej. PodejĹ›cie to, wyrÄłĹĽniaĹ‚o siÄ™ caĹ‚kowitym brakiem dynamiki dziaĹ‚ania systemu internetowego, ponieważ kaĹĽde zdarzenie wywoĹ‚ywane przez oprogramowanie klienta, wymagaĹ‚o zaadresowania i wygenerowania nowego ĹĽÄ…dania w kierunku serwera, ktÄłrego odpowiedziÄ… byĹ‚a nowa zawartoĹ›Ä‡ warstwy prezentacyjnej systemu.

W związku z zauważeniem pewnej regularnoĹ›ci dotyczÄ…cej funkcjonowania wiÄ™kszoĹ›ci systemÄłw internetowych, związanej z faktem niejednokrotnego generowania nieznacznie rÄłĹĽniÄ…cych siÄ™ od siebie odpowiedzi serwera, a takĹĽe w związku z rozwojem jÄ™zyka skryptowego JavaScript oraz technologii Flash, aplikacje w ramach architektury monolitycznej zaczÄ™Ĺ‚y uwzglÄ™dniaÄ‡ obsĹ‚ugÄ™ ĹĽÄ…daĹ„, zawierajÄ…cych przetworzone fragmenty warstwy prezentacyjnej. Ponadto, moĹĽliwa staĹ‚a siÄ™ dynamiczna podmiana okreĹ›lonych fragmentÄłw treĹ›ci, bez koniecznoĹ›ci ponownego pozyskiwania pozostaĹ‚ej zawartoĹ›ci widoku. Usprawnienie to, opierajÄ…ce siÄ™ na technice realizacji ĹĽÄ…daĹ„ asynchronicznych w ramach JavaScript (*ang. AJAX - Asynchronous JavaScript and XML*) pozwoliĹ‚o na poprawÄ™ wydajnoĹ›ci działania aplikacji internetowych przyczyniajÄ…c siÄ™ do zmniejszenia czÄ™stotliwoĹ›ci generowania zapytaĹ„, a takĹĽe redukcji rozmiaru pojedynczej odpowiedzi serwera. RozwiÄ…zanie to, nie wpĹ‚ywaĹ‚o jednakĹĽe bezpoĹ›rednio na strukturÄ™ systemu, ktÄłrej gĹ‚Ählnymi mankamentami byĹ‚y: pojedynczy centralny punkt przetwarzania ĹĽÄ…daĹ„, a takĹĽe brak separacji logiki działania systemu od warstwy prezentacyjnej.

NiedoskonaĹ‚oĹ›ci omÄłwionego powyĹĽej modelu zostaĹ‚y zniwelowane poprzez wprowadzenie architektury zorientowanej na serwisy (*ang. SOA - Service Oriented Architecture*). W podejĹ›ciu tym, dokonano separacji warstwy prezentacyjnej systemu, a takĹĽe wszystkich pozostaĹ‚ych funkcjonalnoĹ›ci dotyczÄ…cych logiki biznesowej oraz przetwarzania danych. ReuĹĽywalne oraz autonomiczne usĹ‚ugi sieciowe pozwalaĹ‚y na realizacjÄ™ okreĹ›lonych funkcji systemu, a sposÄłb komunikacji klienta z usĹ‚ugÄ…, jak i komunikacji pomiÄ™dzy poszczegÄłlnymi serwisami definiowany byĹ‚ przez standaryzowane kontrakty. Zdefiniowanie architektury zorientowanej na serwisy umoĹĽliwiĹ‚o budowÄ™ skalowalnych systemÄłw internetowych, ktÄłrych poszczegÄłlne czÄ™Ĺ›ci mogĹ‚y byÄ‡ realizowane w dowolnej technologii, a implementacja nowej funkcjonalnoĹ›ci nie wymagaĹ‚a przebudowy pozostaĹ‚ych komponentÄłw. RozwiÄ…zanie to, wprowadzaĹ‚o jednak dodatkowy narzut dla kaĹĽdej z przesyĹ‚anych wiadomoĹ›ci, wynikajÄ…cy ze Ĺ›ciĹ›le okreĹ›lonej struktury ĹĽÄ…dania, tworzonej z wykorzystaniem jÄ™zyka XML (*ang. Extensible Markup Language*). Ponadto, wraz ze wzrostem poziomu zaawansowania systemu internetowego, autonomicznoĹ›Ä‡ oraz reuĹĽywalnoĹ›Ä‡ poszczegÄłlnych komponentÄłw malaĹ‚a ze względu na powstawanie specyficznych dla okreĹ›lonego rozwiÄ…zania zaleĹĽnoĹ›ci [**?**].

W zwiÄ…zku z coraz to wiÄ™kszymi wymaganiami dotyczÄ…cymi aplikacji internetowych, dominujÄ…ca Ăławcześ›nie architektura rozproszonych usĹ‚ug sieciowych zastÄ…piona zostaĹ‚a poprzez model uwzglÄ™dniajÄ…cy warstwÄ™ klienckÄ… oraz interfejs programowania aplikacji (*ang. Application Programming Interface*). W przypadku nowoczesnych systemĂłw internetowych, oba z tych komponentĂłw budowane sÄ… w oparciu o architekturÄ™ n-warstwowÄ… (*ang. N-Tier Architecture Application*). W ramach niniejszego modelu, klient wysyĹ‚a ĹĽÄ…danie do interfejsu API, ktĂłry na poczÄ…tku przetwarza jego treś›Ä‡, a nastÄ™pnie wywoĹ‚uje usĹ‚ugÄ™ utworzonÄ… w celu realizacji okreś›lonego zadania. Celem serwisu jest przetworzenie logiki biznesowej dla danej funkcjonalnoś›ci, a takĹĽe odwoĹ‚anie siÄ™ do usĹ‚ug dostÄ™pu do danych w celu ich uzyskania z zewnÄ™trznego ĹşrĂłdĹ‚a informacji. Odpowiednio przygotowana odpowiedĹş jest nastÄ™pnie przekazywana do warstwy obsĹ‚ugi ĹĽÄ…dania, ktĂłra zwraca jÄ… okreś›lonemu klientowi. W przeciwieĹ„stwie do pierwszego z przytoczonych modeli, odpowiedziÄ… API nie jest dokument HTML, a jedynie dane dotyczÄ…ce zasobu, ktĂłre chce uzyskaÄ‡ klient. Sam zasĂłb natomiast, nie jest elementem warstwy prezentacji systemu a zbiorem danych lub typem operacji, ktĂłre moĹĽna na tym zbiorze wykonaÄ‡. UpraszczajÄ…c, stwierdziÄ‡ moĹĽna, ĹĽe API peĹ‚ni rolÄ™ poś›rednika pomiÄ™dzy warstwÄ… prezentacji a zbiorem danych oraz operacji ich przetwarzania, a takĹĽe dostarczania. PoszczegĂłlne usĹ‚ugi realizujÄ…ce logikÄ™ biznesowÄ… aplikacji zawarte sÄ… bezpoś›rednio wewnÄ…trz API, co nie oznacza jednakĹĽe, ĹĽe nie mogÄ… odwoĹ‚ywaÄ‡ siÄ™ do serwisĂłw zewnÄ™trznych. Takie podejś›cie do budowania systemĂłw internetowych zapewnia zarĂłwno skalowalnoś›Ä‡ poszczegĂłlnych aplikacji wchodzÄ…cych w skĹ‚ad systemu, jak i rozwiÄ…zuje problemy architektury SOA zwiÄ…zane z zaleĹĽnoś›ciami wystÄ™pujÄ…cymi pomiÄ™dzy usĹ‚ugami. Dlatego teĹĽ, architektura ta jest powszechnie wykorzystywana w celu budowy i zarzÄ…dzania nowoczesnymi oraz zaawansowanymi systemami internetowymi [**?**].

ZarĂłwno zdecentralizowana architektura zorientowana na serwisy, jak i centralna architektura oparta o interfejs programowania aplikacji, w przeciwieĹ„stwie do architektury monolitycznej, dostarcza zdecydowanie wiÄ™cej moĹĽliwoś›ci zwiÄ…zanych z ewaluacjÄ… dziaĹ‚ania poszczegĂłlnych komponentĂłw systemu. DziÄ™ki powstaniu ostatnich dwĂłch, spoś›rĂłd trzech przedstawionych modeli architektonicznych, moĹĽliwe jest nie tylko zbudowanie efektywnie dziaĹ‚ającej aplikacji internetowej, ale takĹĽe ciÄ…gĹ‚a ocena poprawnoś›ci implementacji jej komponentĂłw, w celu ustawicznego doskonalenia caĹ‚ego systemu.

Niniejsza praca, traktowaÄ‡ bÄ™dzie o ewaluacji efektywnoś›ci dziaĹ‚ania interfejsĂłw programowania aplikacji, w kontekś›cie jednych z dwĂłch najpopularniejszych ś›rodowisk rozwoju oraz uruchamiania api. Ponadto, porĂłwnane zostanÄ… parametry wydajnoś›ciowe w kontekś›cie okreś›lonych przypadkĂłw uĹĽycia interfejsu API, bÄ™dÄ…cego niezbÄ™dnÄ… czÄ™ś›ciÄ… powszechnie wykorzystywanej architektury systemĂłw internetowych.

1.2. Cel i zakres pracy

Celem pracy jest porĂłwnanie wydajnoś›ci dziaĹ‚ania interfejsĂłw programowania aplikacji, tworzonych z wykorzystaniem jÄ™zykĂłw programowania C# oraz JavaScript. Interfejsy, wykonywane sÄ… w dwĂłch rĂłĹĽnych ś›rodowiskach uruchomieniowych. Dla jÄ™zyka C#, ś›rodowiskiem tym jest platforma .NET, natomiast dla jÄ™zyka JavaScript â€“ platforma NodeJS. Analiza porĂłwnawcza, obejmuje zarĂłwno aspekty dotyczÄ…ce efektywnoś›ci dziaĹ‚ania samego interfejsu programowania aplikacji, jaki i elementĂłw wchodzÄ…cych w skĹ‚ad tworzonego systemu. Wś›rĂłd omawianych rozwiÄ…zaĹ„ wyrĂłĹĽniÄ‡ naleĹĽy

mappery obiektowo-relacyjne, systemy bazodanowe, czy teĹĽ mechanizmy zarzÄ…dzania pamiÄ™ciÄ… podrÄ™cznÄ…. NiektÄłre spoĹ›rÄłd wymienionych moduĹ,Ăłw stanowiÄ… integralnÄ… czÄ™ĹšÄ‡ API, natomiast pozostaĹ,e sĹ,uĹĽÄ… do rozszerzenia jego funkcjonalnoĹ›ci.

Zakres pracy obejmuje: przeglÄ…d literaturowy, implementacjÄ™ Ĺ›rodowiska badawczego, realizacjÄ™ badaĹ„ oraz opracowanie wynikÄłw. PrzeglÄ…d literatury tyczy siÄ™ aspektÄłw zwiÄ…zanych ze strukturÄ… i zasadÄ… dziaĹ,ania interfejsÄłw programowania aplikacji, a takĹĽe kwestii dotyczÄ…cych wykonywania pomiarÄłw wydajnoĹ›ci dla poszczegÄłlnych operacji sieciowych. Operacje sieciowe, realizowane sÄ… w ramach obsĹ,ugi ĹĽÄ…dania przez API. Etap implementacji Ĺ›rodowisk badawczych skĹ,ada siÄ™ z budowy interfejsÄłw w oparciu o porÄłwnywane Ĺ›rodowiska rozwoju i uruchamiania aplikacji, a takĹĽe konfiguracji platformy lokalnej oraz platform chmurowych, pozwalajÄ…cych na przeprowadzanie analizy dziaĹ,ania systemÄłw. Realizacja badaĹ„ przeprowadzona zostaĹ,a pod kÄ…tem pomiaru czasu odpowiedzi na ĹĽÄ…dania uĹĽytkownika koĹ„cowego biorÄ…c pod uwagÄ™ aspekty: wywoĹ,ania serii ĹĽÄ…daĹ„ obsĹ,ugi wspÄłĹ,bieĹĽnoĹ›ci procesÄłw, dostÄ™pnoĹ›ci zasobÄłw platformy hostingowej, a takĹĽe moĹĽliwoĹ›ci oferowanych przez mappery obiektowo-relacyjne oraz systemy bazodanowe. Celem etapu opracowania wynikÄłw jest przedstawienie, wizualizacja oraz analiza rÄłĹĽnic wartoĹ›ci czasÄłw odpowiedzi interfesjÄłw API na poszczegÄłlne ĹĽÄ…dania, w odniesieniu do przeprowadzonych badaĹ„. Zastosowanymi kryteriami oceny podczas przeprowadzanej analizy jest czas odpowiedzi interfejsu programowania aplikacji dla wygenerowanego ĹĽÄ…dania, a takĹĽe maksymalna liczba ĹĽÄ…daĹ„ jakie jest w stanie obsĹ,uĹĽyÄ‡ okreĹ›lone API. Przedstawione kryteria, uwzglÄ™dniane zostaĹ,y w odniesieniu do wykorzystywanego Ĺ›rodowiska uruchomieniowego oraz technologii implementacyjnej. Przeprowadzone badania, majÄ… sĹ,uĹĽyÄ‡ wskazaniu zarÄłwno pozytywnych aspektÄłw, jak i problemÄłw dotyczÄ…cych wydajnoĹ›ci dziaĹ,ania aplikacji tworzonych z wykorzystaniem porÄłwnywanych technologii. Ponadto, celem jest takĹĽe przedstawienie moĹĽliwoĹ›ci zwiÄ™kszenia efektywnoĹ›ci implementowanych interfejsÄłw programowania aplikacji.

1.3. Struktura pracy

Niniejsza praca, podzielona zostaĹ,a na szeĹ›Ä‡ rozdziaĹ,Ăłw.

Pierwszy z nich, napisany zostaĹ, w celu zobrazowania dziedziny rozwaĹĽanego problemu, a takĹĽe podkreĹ›lenia jego wagi w kontekĹ›cie zagadnienia usĹ,ug sieciowych. Ponadto, w rozdziale tym zdefiniowano cel popeĹ,nionej pracy oraz przedstawiono zakres czynnoĹ›ci realizowanych w ramach przeprowadzonych badaĹ„.

W rozdziale drugim dokonano wprowadzenia teoretycznego do tematyki interfejsÄłw programowania aplikacji oraz testowania usĹ,ug sieciowych. Wprowadzenie to, w odniesieniu do interfejsÄłw API dotyczy zarÄłwno struktury i zasady dziaĹ,ania omawianej usĹ,ugi sieciowej, jak i sposobu realizacji poĹ,Ä…czeĹ„ tej usĹ,ugi z zewnÄ™trznymi ĹşrÄłdĹ,ami danych. W ramach wprowadzenia do tematyki testowania usĹ,ug sieciowych wyjaĹ›niono fundamentalne pojÄ™cia teorii testowania oraz omÄłwiono dostÄ™pne modele realizacji testÄłw. Co wiÄ™cej, nakreĹ›lono strategiÄ™ wykonywania pomiarÄłw wydajnoĹ›ci w kontekĹ›cie usĹ,ug pracujÄ…cych w sieciach komputerowych. W niniejszym rozdziale, zawarto rÄłwnieĹĽ przeglÄ…d pozycji literaturowych, pomocnych w trakcie realizacji badaĹ„, a takĹĽe przeglÄ…d technologii informatycznych, zastosowanych w procesie implementacji Ĺ›rodowiska badawczego oraz wykonania pomiarÄłw.

W ramach trzeciego z rozdziaĹ,ĂŁw, zdefiniowano i omĂŁwiono kaĹĽdy z aspektĂŁw rozwaĹĽanego problemu badawczego. DziÄ™ki temu, moĹĽliwe staĹ,o siÄ™ sformuĹ,owanie zbioru rozwaĹĽanych scenariuszy badawczych.

W celu realizacji badaĹ„, opartych o zdefiniowane w rozdziale trzecim scenariusze badawcze, naleĹĽy zaprojektowaÄ‡ oraz zaimplementowaÄ‡ odpowiednio dostosowane Ĺ›rodowisko badaĹ„. PoszczegĂŁlne kroki realizacji tego Ĺ›rodowiska, zarĂŁwno te, dotyczÄ…ce jego fizycznej struktury, jak i te, ktĂŁre tyczÄ… siÄ™ implementacji interfejsĂŁw programowania aplikacji, opisane zostaĹ,y w rozdziale czwartym niniejszej pracy.

PiÄ…ty z rozdziaĹ,ĂŁw, ma na celu przedstawienie rezultatĂŁw wynikajÄ…cych z przeprowadzonych prac naukowych. Rezultaty te, w obrÄ™bie niniejszego rozdziaĹ,u zostaĹ,y zgrupowane wzglÄ™dem zdefiniowanych uprzednio scenariuszy badawczych, realizowanych w odpowiednio przystosowanym Ĺ›rodowisku. Ponadto, dla uzyskanych wartoĹ›ci pomiarowych, dotyczÄ…cych kryteriĂŁw poszczegĂłlnych badaĹ„, wykonano testy parametryczne, dziÄ™ki ktĂłrym moĹĽliwa jest ocena istotnoĹ›ci statystycznej zaobserwowanych rĂłĹĽnic wynikowych. Co wiÄ™cej, wyniki kaĹĽdego z realizowanych scenariuszy badawczych poddane zostaĹ,y krytycznej analizie.

Ostatni z rozdziaĹ,ĂŁw peĹ,ni rolÄ™ podsumowania. Autor przedstawia w nim uzyskane efekty wykonanej pracy, a takĹĽe nakreĹ›la moĹĽliwoĹ›ci zwiÄ…zane z dalszym rozwojem badaĹ„.

Rozdział 2

Wprowadzenie teoretyczne

2.1. Wykorzystywane terminy

W niniejszej pracy, posŁ,uŁĽono siÄ™ terminologiÄ… dystynktywnÄ… z punktu widzenia realizacji, rozwoju oraz ewaluacji usŁ,ug sieciowych. Najbardziej istotne spoŁ›rĂłd wykorzystywanych terminĂłw wymieniono poniŁĽej. Dla kaŁĽdego z pojÄ™Ä‡, przedstawiono obcojÄ™zyczne tŁ,umaczenie, a takŁĽe zdefiniowano spĂłjny oraz zwiÄ™zŁ,y opis.

UsŁ,uga sieciowa

Web Service

Rodzaj systemu informatycznego cechujÄ…cego siÄ™ permanentnym wykonywaniem zdefiniowanych funkcji, tuŁĽ po uzyskaniu ŁĽÄ…dania. Ł»Ä…danie to, przybiera postaÄ‡ danych, przekazanych w ramach systematycznej struktury. SposĂłb dostarczenia ŁĽÄ…dania, jego format, a takŁĽe metoda odpowiedzi na ŁĽÄ…danie, definiowane sÄ… poprzez protokĂłŁ, sieciowy z ktĂłrego korzysta dana usŁ,uga.

Interfejs Programowania Aplikacji

Application Programming Interface

ZbiĂłr reguŁ, oraz struktur programistycznych okreŁ›lajÄ…cy metodÄ™ oraz cel interakcji pomiÄ™dzy komponentami oprogramowania. PojÄ™cie interfejsu programowania aplikacji jest niezaleŁĽne od warstwy implementacji systemu i moŁĽe tyczyÄ‡ siÄ™ dowolnego rodzaju programu komputerowego. Interfejs API definiowany jest na poziomie kodu ŁşrĂłdŁ,owego poszczegĂłlnych fragmentĂłw oprogramowania, a jego zadaniem jest dostarczenie wymaganych specyfikacji struktur programistycznych, a takŁĽe protokoŁ,u, pozwalajÄ…cego na ich wykorzystanie przez zewnÄ™trzny komponent programowy.

API wykonane w technologii REST

RESTful API

Interfejs programowania aplikacji, bazujÄ…cy zarĂłwno w swojej strukturze, jak i funkcjonalnoŁ›ci na zbiorze Ł›ciŁ›le okreŁ›lonych reguŁ, dostarczanych w ramach metodologii REST. ReguŁ,y metodologii tej, implementowane sÄ… najczÄ™Ł›ciej w stosunku do interfejsĂłw programowania aplikacji, ktĂłre wykorzystujÄ… w kontekŁ›cie zewnÄ™trznej komunikacji protokĂłŁ, hipertekstowy. Podstawowa charakterystyka interfejsu programowania aplikacji

opartego o zbiĂłr reguĹ, REST definiowana jest poprzez bezstanowoĹ›Ä‡ transmisji danych, pojÄ™cia zasobĂłw i reprezentacji, a takĹĽe cechÄ™ jednolitoĹ›ci interfejsu komunikacyjnego.

Kontroler

Controller

Struktura programistyczna (w przypadku jÄ™zykĂłw programowania opartych o paradygmat obiektowy - klasa), ktĂłrej celem jest obsĹ‚uga ĹĽÄ…dania dostarczonego od aplikacji klienckiej, zweryfikowania zgodnoĹ›ci jego treĹ›ci, a nastÄ™pnie przekierowanie wykonania programu do odpowiedniej metody warstwy logiki biznesowej. Po zakoĹ„czeniu wszystkich operacji dotyczÄ…cych omawianego zapytania, metoda dostÄ™pna w ramach klasy kontrolera formuĹ‚uje odpowiedĹş zwracanÄ… bezpoĹ›rednio do konsumenta ĹĽÄ…dania.

Serwis

Service

Struktura programistyczna (w przypadku jÄ™zykĂłw programowania opartych o paradygmat obiektowy - klasa), ktĂłrej zadaniem jest realizacja obliczeĹ„ zwiÄ…zanych z okreĹ›lonym obiektem, lub domenÄ…, w ramach ktĂłrej okreĹ›lony obiekt siÄ™ znajduje. Klasy serwisĂłw (nazywane rĂłwnieĹĽ klasami logiki biznesowej), stanowiÄ… centralny punkt przetwarzania w ramach interfejsĂłw programowania aplikacji, a takĹĽe peĹ‚niÄ… rolÄ™ poĹ›rednika pomiÄ™dzy komponentami odpowiedzialnymi za zarzÄ…dzanie ĹĽÄ…daniem (tj. metodami kontrolerĂłw) oraz pozyskiwanie danych z okreĹ›lonych ĹşrĂłdeĹ‚ (tj. metodami repozytoriĂłw).

Repozytorium

Repository

Struktura programistyczna (w przypadku jÄ™zykĂłw programowania opartych o paradygmat obiektowy - klasa), ktĂłrej rolÄ… jest wykonywanie operacji na obiektach modelu danych komunikujÄ…c siÄ™ bezpoĹ›rednio z wykorzystywanym systemem bazodanowym. Operacje zawarte wewnÄ…trz metod klas repozytoriĂłw mogÄ… mieÄ‡ postaÄ‡ kwerend lub komend definiowanych w jÄ™zyku zapytaĹ„, dostarczanym przez serwer bazodanowy, mogÄ… operowaÄ‡ na dostÄ™pnej w pamiÄ™ci aplikacji strukturze danych, bÄ…dĹş teĹĽ odwoĹ‚ywaÄ‡ siÄ™ do zewnÄ™trznego zbioru bazodanowego, manipulujÄ…c instancjami klas zdefiniowanego wewnÄ…trz aplikacji modelu. W ostatnim z omawianych przypadkĂłw, aktualizacje wartoĹ›ci w ramach encji modelu danych sÄ… identyfikowane przez maper obiektowo-relacyjny, ktĂłry generuje polecenia jÄ™zyka zapytaĹ„, systemu bazodanowego, majÄ…ce na celu synchronizacjÄ™ stanu danych aplikacji oraz zewnÄ™trznego ĹşrĂłdĹ‚a informacji. Metody klas repozytoriĂłw sÄ… wywoĹ‚ywane wewnÄ…trz metod logiki biznesowej.

Mapper obiektowo-relacyjny

Object-relational mapper

Oprogramowanie, ktĂłrego gĹ‚Ăłwnym zadaniem jest konwersja struktury klas modelu danych do fizycznej organizacji tabel w ramach systemu bazodanowego. Ponadto, mapper obiektowo-

relacyjny dostarcza zbiĂłr wĹ,aĹ›ciwoĹ›ci oraz metod stanowiÄ…cych fasadÄ™ dla niskopoziomowych procedur dostÄ™pu do bazy danych, a takĹĽe modyfikacji danych w niej zawartych.

PamiÄ™Ä‡ podrÄ™czna

Cache

Wydzielony fragment pamiÄ™ci cechujÄ…cy siÄ™ szybkim czasem dostÄ™pu, wysokÄ… przepustowoĹ›ciÄ… transmisji, a takĹĽe ograniczonym okresem trwaĹ,ego przechowywania danych. PamiÄ™Ä‡ ta, w kontekĹ›cie webowego interfejsu programowania aplikacji, wykorzystywana jest w celu przechowywania wynikĂłw czÄ™sto realizowanych operacji, a takĹĽe magazynowania uprzednio dostarczonych do klienta fragmentĂłw odpowiedzi na ĹĽÄ…dania.

Przetwarzanie wspĂłĹ,bieĹĽne

Concurrent Computing

Technika programistyczna oparta o wykorzystanie wielu wspĂłĹ,istniejÄ…cych procesĂłw oraz wÄ…tkĂłw, dostÄ™pnych w obrÄ™bie jednej aplikacji, a takĹĽe odwoĹ,ujÄ…cych siÄ™ do wspĂłĹ,dzielonych struktur danych. PoszczegĂłlne wÄ…tki stanowiÄ… elementy skĹ,adowe pojedynczego procesu i sÄ… uruchamiane na tej samej centralnej jednostce przetwarzania. Procesor wykonuje operacje przeĹ,Ä…czania pomiÄ™dzy kontekstami poszczegĂłlnych wÄ…tkĂłw, dziÄ™ki czemu uzyskaÄ‡ moĹĽna wraĹĽenie, wykonania wielu spoĹ›rĂłd tych elementĂłw w sposĂłb rĂłwnolegĹ,y. Z punktu widzenia interfejsu programowania aplikacji, ktĂłry implementuje technikÄ™ przetwarzania wspĂłĹ,bieĹĽnego, wyrĂłĹĽniÄ‡ moĹĽemy punkty koĹ„cowe, ktĂłre definiowane sÄ… jako osobne wÄ…tki aplikacji. Dlatego teĹĽ, interfejs programowania aplikacji cechuje siÄ™ dostÄ™pnoĹ›ciÄ… pomimo jednoczesnego przetwarzania wielu, niekiedy dĹ,ugotrwaĹ,ych ĹĽÄ…daĹ„ klientĂłw.

Algorytm metaheurystyczny

Metaheuristic algorithm

Technika projektowania algorytmĂłw nie zapewniajÄ…cych gwarancji uzyskania optimum dla rozwaĹĽanego problemu, jednakĹĽe pozwalajÄ…ca na zbudowanie systemu, dostarczajÄ…cego rozwiÄ…zanie zĹ,oĹĽonego zagadnienia w akceptowalnym czasie, a takĹĽe uzyskiwanego przy wykorzystaniu akceptowalnej iloĹ›ci zasobĂłw sprzÄ™towych. Algorytm metaheurystyczny, poza konwencjonalnymi reguĹ,ami stosowanymi w ramach standardowych wzorcĂłw programowania, implementuje reguĹ,y rozwiÄ…zywania problemĂłw oparte na losowoĹ›ci, bÄ…dĹş teĹĽ wywnioskowane na podstawie zjawisk fizycznych.

Punkt koĹ„cowy interfejsu programowania aplikacji

API Endpoint

Punkt koĹ„cowy usĹ,ugi sieci web definiuje jeden z koĹ„cĂłw kanaĹ,u komunikacyjnego pomiÄ™dzy aplikacjÄ… klienckÄ… a serwerowÄ…. W momencie interakcji interfejsu programowania aplikacji z odrÄ™bnym systemem, punkt styku dwĂłch usĹ,ug sieciowych w ramach omawianej interakcji nazywany jest punktem koĹ„cowym. W odniesieniu do wewnÄ™trznej struktury interfejsu programowania aplikacji, punkt koĹ„cowy wywoĹ,uje zwiÄ…zanÄ… z nim metodÄ™ klasy kontrolera, a samo powiÄ…zanie identyfikowane jest (w przypadku

usĹ‚ugi sieciowej bazujÄ…cej na protokole hipertekstowym i metodologii REST) poprzez nazwÄ™ zasobu, rodzaj metody, a takĹĽe parametry ĹĽÄ…dania.

Ĺ»Ä…danie realizowane w ramach usĹ‚ugi protokoĹ‚u hipertekstowego

HTTP Request

Struktura danych, wysyĹ‚ana od aplikacji klienckiej (tj. aplikacji internetowej, przeglÄ…darki, czy teĹĽ programu klienta HTTP) w kierunku usĹ‚ugi sieciowej. Ĺ»Ä…danie protokoĹ‚u hipertekstowego charakteryzuje siÄ™ jednoznacznie zdefiniowanÄ… strukturÄ…, uwzglÄ™dniajÄ…cÄ… m.in. unikalny identyfikator zasobu, listÄ™ zdefiniowanych nagĹ‚ÄĄwkÄĄw, ciaĹ‚o ĹĽÄ…dania oraz jednÄ… z dziewiÄ™ciu dopuszczalnych metod HTTP.

OdpowiedĹş usĹ‚ugi protokoĹ‚u hipertekstowego

HTTP Response

Struktura danych, wysyĹ‚ana przez usĹ‚ugÄ™ sieciowÄ… w kierunku aplikacji klienckiej. OdpowiedĹş HTTP, ma na celu poinformowanie klienta serwisu webowego o statusie realizacji, wysĹ‚anego przez niego uprzednio ĹĽÄ…dania. Podstawowymi elementami odpowiedzi usĹ‚ugi protokoĹ‚u hipertekstowego sÄ…: ciaĹ‚o odpowiedzi (zdefiniowane najczÄ™Ĺ›ciej z wykorzystaniem notacji JSON lub jÄ™zyka XML), kod odpowiedzi (liczba determinujÄ…ca stan wykonania ĹĽÄ…dania), a takĹĽe zbiÄĄr informacji nagĹ‚ÄĄwkowych dotyczÄ…cych typu danych zawartych w odpowiedzi, czy teĹĽ fizycznych informacji o serwerze usĹ‚ugi sieciowej.

Kod odpowiedzi usĹ‚ugi protokoĹ‚u hipertekstowego

HTTP Response Code

Liczba determinujÄ…ca status realizacji ĹĽÄ…dania wysĹ‚anego przez aplikacjÄ™ klienckÄ…. Kod odpowiedzi stanowi jednÄ… z wymaganych skĹ‚adowych dotyczÄ…cych standardowego rezultatu zwracanego w ramach usĹ‚ugi opartej o protokÄŁĹ‚, hipertekstowy. WyrÄŁĹĽniÄ‡ moĹĽemy piÄ™Ä‡ kategorii kodÄŁw odpowiedzi, niosÄ…cych ze sobÄ… odmiennÄ… informacje. Kategoriami tymi sÄ…: kody informacyjnej odpowiedzi (100-199), kody poprawnej odpowiedzi (200-299), kody wiadomoĹ›ci o przekierowaniu (300-399), kody bĹ‚Ä™du aplikacji klienckiej (400-499), oraz kody bĹ‚Ä™du aplikacji serwerowej (500-599).

Czas odpowiedzi usĹ‚ugi protokoĹ‚u hipertekstowego

HTTP Response Time

WyraĹĽony w milisekundach, przedziaĹ‚ czasu od momentu otrzymania ĹĽÄ…dania wygenerowanego przez aplikacjÄ™ klienckÄ…, do chwili zwrÄŁcenia rezultatu wykonywanych przez usĹ‚ugÄ™ sieciowÄ… obliczeĹ„. Liczba ta, stanowi jednÄ… z wartoĹ›ci pomiarowych, w kontekĹ›cie efektywnoĹ›ci dziaĹ‚ania interfejsu programowania aplikacji.

Obiektowa notacja JavaScript

JavaScript Object Notation - JSON

Format definicji, reprezentacji, a takĹĽe wymiany danych w postaci obiektĂłw niezaleĹĽny od okreĹ›lonego jÄ™zyka programowania. Obiektowa notacja JavaScript jest powszechnie wykorzystywana jako format komunikatĂłw przekazywanych pomiÄ™dzy interfejsami programowania aplikacji a systemami klienckimi. W odrĂłĹĽnieniu od jÄ™zyka reprezentacji danych opartego o znaczniki *Extensible Markup Language - XML*, obiektowa notacja JavaScript cechuje siÄ™ mniejszym rozmiarem przesyĹ‚anych obiektĂłw (poprzez redukcjÄ™ liczby metadanych), jednolitym standardem niezaleĹĽnym od technologii, a takĹĽe brakiem przechowywania informacji o typie poszczegĂłlnej wartoĹ›ci zadanego obiektu.

Testy wzorcowe

Benchmark

Rodzaj ewaluacji oprogramowania, ktĂłrej zadaniem jest okreĹ›lenie referencyjnego poziomu wydajnoĹ›ci dla testowanego systemu. Metryki, uzyskane w ramach testĂłw wzorcowych, mogÄ… zostaÄ‡ wykorzystane jako wartoĹ›ci ograniczeĹ„ wzglÄ™dem testĂłw obciÄ…ĹĽeniowych oraz przeciÄ…ĹĽeniowych.

Testy dymne

Smoke testing

Metoda testowania oprogramowania, ktĂłrej celem jest sprawdzenie poprawnoĹ›ci funkcjonowania poszczegĂłlnych elementĂłw systemu. Testy dymne, wykonywane sÄ… przed testami wydajnoĹ›ciowymi, po to aby upewniÄ‡ siÄ™ co do braku bĹ‚Ä™dĂłw implementacyjnych w ramach analizowanego oprogramowania.

Testy wydajnoĹ›ci podstawowej

Baseline performance testing

Metoda ewaluacji oprogramowania, pozwalajÄ…ca na weryfikacjÄ™ dziaĹ‚ania systemu w warunkach analogicznych do realiĂłw standardowego dziaĹ‚ania. Na podstawie testĂłw wydajnoĹ›ci podstawowej, okreĹ›liÄ‡ moĹĽna wartoĹ›ci metryk, ktĂłre bÄ™dÄ… miaĹ‚y zastosowanie jako punkt odniesienia dla kolejnych rodzajĂłw testĂłw. Ponadto, wykorzystujÄ…c standard pomiaru wydajnoĹ›ci aplikacji internetowych (taki jak np. APDEX), wartoĹ›ci uzyskane w ramach ewaluacji podstawowych, mogÄ… posĹ‚uĹĽyÄ‡ w celu okreĹ›lenia punktĂłw satysfakcji, tolerancji oraz frustracji.

Testy obciÄ…ĹĽajÄ…ce

Load testing

Rodzaj testĂłw, ktĂłre majÄ… na celu okreĹ›lenie maksymalnego poziomu natÄ™ĹĽenia operacji, jakie mogÄ… byÄ‡ generowane w kierunku oprogramowania. W kontekĹ›cie niniejszej pracy, operacjami tymi sÄ… ĹĽÄ…dania wysyĹ‚ane do interfejsu programowania aplikacji. Kluczowym aspektem testu obciÄ…ĹĽeniowego jest zdefiniowanie progu obciÄ…ĹĽenia aplikacji, powyĹĽej ktĂłrego system jest nie w stanie generowaÄ‡ poprawnych odpowiedzi w akceptowalnym czasie.

Testy przeciÄ…ĹĽeniowe

Stress testing

Metoda ewaluacji oprogramowania, w ramach ktĂłrej natÄ™ĹĽenie operacji generowanych w kierunku testowanego oprogramowania zwiÄ™kszone jest ponad ustalony prĂłg tolerancji. Celem testu przeciÄ…ĹĽeniowego jest obserwacja sposobu dziaĹ‚ania systemu, w momencie, w ktĂłrym nie jest on w stanie przetwarzaÄ‡ otrzymywanych ĹĽÄ…daĹ„ w sposĂłb poprawny.

Asercja

Assertion

WyraĹĽenie typu prawda/faĹ‚sz, zdefiniowane w dowolnym miejscu programu, ktĂłre przyjmuje wartoĹ›Ä‡ prawdziwÄ… w momencie speĹ‚nienia hipotezy zawartej w ramach okreĹ›lonego przypadku testowego. Praktyczne podejĹ›cie do procesu testowania funkcjonalnoĹ›ci oprogramowania, sprowadza siÄ™ do definiowania hipotez oraz ciÄ…gĂłw operacji w kontekĹ›cie przypadkĂłw testowych, a nastÄ™pnie weryfikacji tych hipotez z wykorzystaniem asercji.

2.2. Interfejsy programowania aplikacji

Webowy interfejs programowania aplikacji to usĹ‚uga sieciowa, ktĂłrej celem jest realizacja zadaĹ„ zleconych przez oprogramowanie klienta. Zadania te, dotyczÄ… operacji wykonywanych w kontekĹ›cie okreĹ›lonych zasobĂłw. WyrĂłĹĽniÄ‡ moĹĽemy operacje zwane zapytaniami (tj. dotyczÄ…ce pozyskiwania danych z ich ĹşrĂłdeĹ‚) a takĹĽe komendami (tj. zwiÄ…zane z wykonywaniem operacji na danych).

Interfejsy API, budowane sÄ… z wykorzystaniem protokoĹ‚u HTTP, dlatego teĹĽ w ich kontekĹ›cie moĹĽemy mĂłwiÄ‡ o komunikacji bezstanowej definiujÄ…cej pojÄ™cia ĹĽÄ…dania oraz odpowiedzi. W zwiÄ…zku z charakterystykÄ… protokoĹ‚u hipertekstowego, zarĂłwno ĹĽÄ…danie jak i odpowiedĹş cechuje siÄ™ regularnÄ… strukturÄ… zawierajÄ…cÄ… predefiniowane elementy.

Ĺ»Ä…danie protokoĹ‚u http wysyĹ‚ane jest od aplikacji klienta do interfejsu API. PodstawowÄ… skĹ‚adowÄ… tego polecenia stanowi unikalny identyfikator zasobu URI (*ang. Uniform Resource Identifier*), na podstawie ktĂłrego moĹĽliwe jest okreĹ›lenie fragmentu dziedziny obsĹ‚ugiwanego modelu danych. Informacja ta jednak, nie jest wystarczajÄ…ca w kontekĹ›cie realizacji jednej z funkcjonalnoĹ›ci, zdefiniowanych w ramach API. Ĺ»Ä…danie klienta, musi zostaÄ‡ uzupeĹ‚nione o jednÄ… z dziewiÄ™ciu ustalonych metod http, obsĹ‚ugiwanÄ… wersjÄ™ protokoĹ‚u, a takĹĽe zbiĂłr linii nagĹ‚Ăłwkowych. Opcjonalnie, informacja wysyĹ‚ana w kierunku interfejsu, moĹĽe zostaÄ‡ wzbogacona o zawartoĹ›Ä‡ tekstowÄ… okreĹ›lanÄ… ciaĹ‚em ĹĽÄ…dania (*ang. Request body*). Taki zbiĂłr informacji, pozwala na jednoznacznÄ… identyfikacje fragmentu kodu programu, ktĂłry ma zostaÄ‡ wykonany wewnÄ…trz interfejsu programowania aplikacji. W tabelach **??** oraz **??** przedstawiono kolejno listÄ™ zdefiniowanych metod protokoĹ‚u hipertekstowego wraz z wyjaĹ›nieniem ich przeznaczenia, a takĹĽe zbiĂłr najczÄ™Ĺ›ciej wykorzystywanych linii nagĹ‚Ăłwkowych, w kontekĹ›cie realizacji ĹĽÄ…daĹ„.

Po wykonaniu kodu programu przypisanego do okreĹ›lonego rodzaju polecenia generowanego przez aplikacje klienckÄ…, z interfejsu programowania aplikacji zwracana jest odpowiedĹş na ĹĽÄ…danie (*ang. HTTP response*). Analogicznie do instrukcji realizacji danej czynnoĹ›ci, takĹĽe odpowiedĹş dotyczÄ…ca statusu jej wykonania jest ustrukturyzowana zgodnie

Tab. 2.1: ZbiĂłr dozwolonych metod protokoĹ‚u hipertekstowego

Nazwa metody	Opis
GET	Pozyskanie danych dotyczÄ…cych pojedynczej instancji okreĹ›lonego zasobu lub grupy instancji z opcjonalnym uwzglÄ™dnieniem warunkĂłw kwalifikacji poszczegĂłlnej instancji do grupy.
POST	Definiowanie nowej instancji dotyczÄ…cej okreĹ›lonego typu zasobu. Przy zastosowaniu metody POST, wymagane jest zdefiniowanie ciaĹ‚a ĹĽÄ…dania, jako czÄ™Ĺ›ci skĹ‚adowej generowanej instrukcji.
PUT	Aktualizacja peĹ‚ni zawartoĹ›ci instancji wystÄ™pujÄ…cej w ramach odwoĹ‚ania siÄ™ do okreĹ›lonego zasobu. Przy zastosowaniu metody PUT, wymagane jest zdefiniowanie ciaĹ‚a ĹĽÄ…dania, jako czÄ™Ĺ›ci skĹ‚adowej generowanej instrukcji.
DELETE	UsuniÄ™cie istniejÄ…cej instancji dotyczÄ…cej okreĹ›lonego typu zasobu.
PATCH	Aktualizacja fragmentu zawartoĹ›ci instancji wystÄ™pujÄ…cej w ramach odwoĹ‚ania siÄ™ do okreĹ›lonego zasobu. Przy zastosowaniu metody PATCH, wymagane jest zdefiniowanie ciaĹ‚a ĹĽÄ…dania, jako czÄ™Ĺ›ci skĹ‚adowej generowanej instrukcji.
HEAD	Pozyskanie zbioru linii nagĹ‚Ăłwkowych, ktĂłre byĹ‚yby dostarczone wraz z ciaĹ‚em odpowiedzi w ramach ĹĽÄ…dania wykorzystujÄ…cego metodÄ™ GET. Wygenerowanie ĹĽÄ…dania HEAD umoĹĽliwia okreĹ›lenie charakteru danych, przed ich ewentualnym pozyskaniem.
OPTIONS	Pozyskanie informacji dotyczÄ…cych charakterystyki oraz struktury serwera. DefiniujÄ…c ĹĽÄ…danie typu OPTIONS, klient moĹĽe dowiedzieÄ‡ siÄ™ o dopuszczalnych metodach HTTP obsĹ‚ugiwanych przez serwer, czy teĹĽ uzyskaÄ‡ informacje o nazwie serwera oraz wykorzystywanym systemie operacyjnym.
CONNECT	Ustanowienie dwukierunkowej komunikacji pomiÄ™dzy klientem a serwerem. W przypadku realizacji komunikacji szyfrowanej, ĹĽÄ…danie typu CONNECT pozwala na zestawienie zabezpieczonego tunelu pomiÄ™dzy hostami.
TRACE	Wygenerowanie komunikatu diagnostycznego w ramach pÄ™tli zwrotnej, ktĂłrego celem jest osiÄ…gniÄ™cie kaĹĽdego z hostĂłw, biorÄ…cych udziaĹ‚, w komunikacji.

Tab. 2.2: ZbiĂłr najczÄ™Ĺ›ciej wykorzystywanych linii nagĹ,Ăłwkowych w kontekĹ›cie ĹĽÄ…dania protokoĹ,u hipertekstowego

Linia nagĹ,Ăłwkowa	Znaczenie	Dopuszczalna zawartoĹ›Ä‡
accept	Typ zawartoĹ›ci, ktĂłrÄ… jest w stanie przetwarzaÄ‡ aplikacja kliencka	Identyfikator typu MIME (*ang. Multipurpose Internet Mail Extensions*) lub zapis */* oznaczający dowolnÄ… zawartoĹ›Ä‡
accept-encoding	SposĂłb kodowania znakĂłw, rozumiany przez stronÄ™ klienta	ZbiĂłr formatĂłw kodowania zdefiniowany w ramach rejestru formatĂłw IANA
accept-language	JÄ™zyk naturalny, preferowany przez stronÄ™ klienckÄ…	Pojedyncza wartoĹ›Ä‡ reprezentującą okreĹ›lony kraj lub region, bÄ…dĹş teĹĽ lista niniejszych wartoĹ›ci wraz z parametrem istotnoĹ›ci poszczegĂłlnego kodu lokalizacji
content-length	DĹ,ugoĹ›Ä‡ ciaĹ,a ĹĽÄ…dania wyraĹĽona w bajtach	Liczba naturalna
content-type	Format zawartoĹ›ci ciaĹ,a ĹĽÄ…dania	Identyfikator typu MIME wraz ze sposobem kodowania wiadomoĹ›ci
cookie	ZbiĂłr informacji pozwalających na wprowadzenie oraz utrzymanie stanowego charakteru transmisji	Zestaw par klucz-wartoĹ›Ä‡, gdzie klucz jest wartoĹ›ciÄ… tekstowÄ…, a wartoĹ›Ä‡ przyjmuje postaÄ‡ dowolnÄ…
origin	Informacja determinującą pochodzenie ĹĽÄ…dania	CiÄ…g tekstowy skĹ,adający siÄ™ z nazwy protokoĹ,u, nazwy hosta oraz numeru portu
user-agent	Specyfikacja techniczna oprogramowania klienta	CiÄ…g znakĂłw zawierającą informacje o nazwie produktu, jego wersji, platformie sprzÄ™towej, czy teĹĽ systemie operacyjnym

z wytycznymi zawartymi w definicji protokoĹ,u hipertekstowego. W ramach rezultatu zwrĂłconego przez API wyrĂłĹĽniĂ‡ naleĹĽy: adres docelowy klienta, kod statusu, ciaĹ,o odpowiedzi, a takĹĽe zbiĂłr linii nagĹ,Ăłwkowych. Informacja zawarta w ramach kodu statusu, determinuje powodzenie realizowanej operacji, a treĹ›Ă‡ dostarczanych linii nagĹ,Ăłwkowych, moĹĽe zostaĂ‡ wykorzystana w celu wnioskowania o charakterystyce odbywajĂ…cej siĂ™ komunikacji. CiaĹ,o odpowiedzi powinno zawieraĂ‡ dane dotyczĂ…ce definiowanego w ramach identyfikatora ĹĽĂ…dania zasobu, w przypadku ĹĽĂ…daĹ„ wykorzystujĂ…cych metodĂ™ GET. W kontekĹ›cie pozostaĹ,ych ĹĽĂ…daĹ„ zgodnie z wytycznymi dokumentu RFC (*ang. Request For Comments*) o numerze 7230, powinno ono posiadaĂ‡ charakter informacji pomocniczej, bĂ…dĹş teĹĽ pozostaĂ‡ puste [**?**]. W ramach tabel **??** oraz **??**, wymienione zostaĹ,y kolejno: zbiĂłr najczĂ™Ĺ›ciej zwracanych linii nagĹ,Ăłwkowych w kontekĹ›cie odpowiedzi na ĹĽĂ…danie, a takĹĽe przedziaĹ,y liczbowe dla kodĂłw statusu odpowiedzi, wraz z ich semantykĂ….

Tab. 2.3: ZbiĂłr najczĂ™Ĺ›ciej zwracanych linii nagĹ,Ăłwkowych w kontekĹ›cie odpowiedzi protokoĹ,u hipertekstowego

Linia nagĹ,Ăłwkowa	Znaczenie	Dopuszczalna zawartoĹ›Ă‡
access-control-allow-credentials	OkreĹ›lenie, czy odpowiedĹş serwera ma byĂ‡ osiĂ…galna z kodu JavaScript aplikacji klienckiej, w momencie gdy nagĹ,Ăłwek ĹĽĂ…dania dotyczĂ…cy poĹ›wiadczeĹ„, zezwala na ich doĹ,Ă…czenie	WartoĹ›Ă‡ prawda/faĹ,sz
access-control-allow-origin	Informacja dotyczĂ…ca pochodzenia klienta, ktĂłry moĹĽe ubiegaĂ‡ siĂ™ o otrzymanie odpowiedzi od serwera	adres hosta klienckiego lub symbol gwiazdki oznaczajĂ…cy zezwolenie dla wszystkich hostĂłw
cache-control	Dane konfiguracyjne dotyczĂ…ce obsĹ,ugi pamiĂ™ci podrĂ™cznej	ZbiĂłr par klucz-wartoĹ›Ă‡ okreĹ›lajĂ…cych zachowanie pamiĂ™ci cache w kontekĹ›cie okreĹ›lonej komunikacji
content-length	DĹ,ugoĹ›Ă‡ ciaĹ,a odpowiedzi wyraĹĽona w bajtach	Liczba naturalna
content-type	Format zawartoĹ›ci ciaĹ,a odpowiedzi	Identyfikator typu MIME wraz ze sposobem kodowania wiadomoĹ›ci
cross-origin-resource-policy	Polecenie ignorowania ĹĽĂ…daĹ„ realizowanych pomiĂ™dzy ĹşrĂłdĹ,ami bĂ…dĹş witrynami w kontekĹ›cie okreĹ›lonego zasobu	WartoĹ›Ă‡ prawda/faĹ,sz
expires	Data wygaĹ›niĂ™cia waĹĽnoĹ›ci niniejszej odpowiedzi	OkreĹ›lona data
server	Nazwa hosta dostarczajĂ…cego odpowiedĹş klientowi	CiĂ…g znakĂłw

Przedstawiony w niniejszy sposĂłb interfejs programowania aplikacji scharakteryzowaĂ‡ naleĹĽy jako deterministyczny system wejĹ›ciowo-wyjĹ›ciowy. Ponadto, naleĹĽy zauwa-

Tab. 2.4: ZbiĂłr kodĂłw statusu odpowiedzi protokoĹ,u hipertekstowego

PrzedziaĹ, liczbowy	Semantyka w kontekĹ›cie odpowiedzi
100 - 199	ZbiĂłr kodĂłw informacyjnych - ĹĽÄ…danie jest aktualnie przetwarzanie
200 - 299	ZbiĂłr kodĂłw poprawnej odpowiedzi - wystosowane ĹĽÄ…danie zostaĹ,o zrealizowane poprawnie
300 - 399	ZbiĂłr kodĂłw przekierowaĹ„ - istnieÄ‡ moĹĽe wiele akceptowalnych odpowiedzi dla ĹĽÄ…dania bÄ…dĹş realizacja okreĹ›lonej operacji wymusza odwoĹ,anie siÄ™ pod adres identyfikujÄ…cy odmienny zasĂłb
400 - 499	ZbiĂłr kodĂłw bĹ,Ä™du po stronie klienta - wygenerowane ĹĽÄ…danie zawiera bĹ,Ä™dy, oczekiwany zasĂłb nie istnieje, klient nie jest uwierzytelniony lub nie posiada okreĹ›lonego poziomu uprawnieĹ„
500 - 599	ZbiĂłr kodĂłw bĹ,Ä™du po stronie serwera - pomimo poprawnej struktury wygenerowanego ĹĽÄ…dania, serwer nie jest w stanie zrealizowaÄ‡ przydzielonej mu operacji

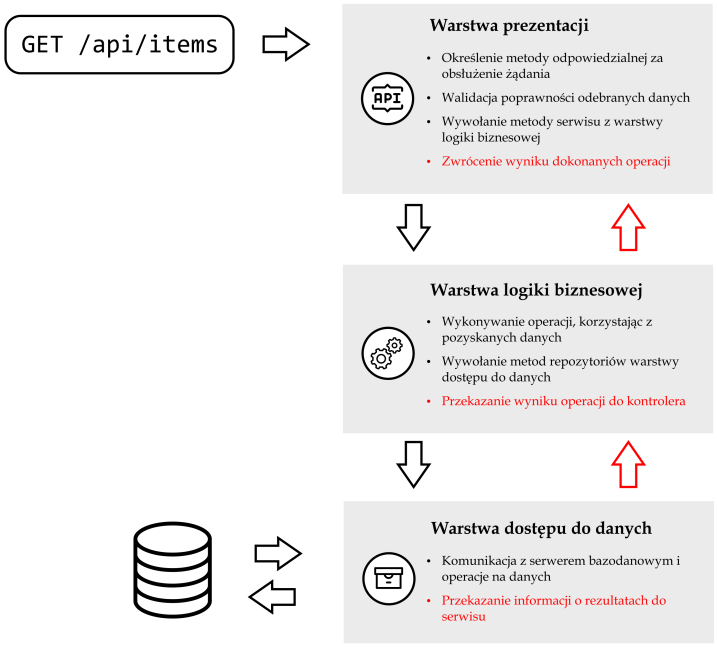
ĹĽyÄ‡, ĹĽe w ramach systemu tego, wystÄ™puje zjawisko inercji, powodowane koniecznoĹ›ciÄ… realizacji zdefiniowanego w ramach API kodu programu. Na podstawie tego zaĹ,oĹĽenia, ewaluacjÄ™ dziaĹ,ania oraz wydajnoĹ›ci interfejsu programowania aplikacji przeprowadziÄ‡ moĹĽna poprzez wprowadzanie okreĹ›lonego wejĹ›cia (tj. generowanie ĹĽÄ…dania) oraz obserwacjÄ™ wartoĹ›ci zwrĂłconej na wyjĹ›ciu (tj. uzyskana odpowiedĹş).

Proces przetwarzania ĹĽÄ…dania wewnÄ…trz interfejsu API

Po uzyskaniu ĹĽÄ…dania otrzymanego od strony klienta, zadaniem interfejsu programowania aplikacji jest wybĂłr okreĹ›lonej klasy kontrolera, a takĹĽe zawartej w niej metody. KaĹĽda z klas kontrolerĂłw stworzona jest w celu obsĹ,ugi operacji zwiÄ…zanych z konkretnym zasobem, a poszczegĂłlna metoda tej klasy implementuje zachowanie ktĂłre ma zostaÄ‡ wywoĹ,ane w kontekĹ›cie dostarczonego typu oraz identyfikatora polecenia.

WewnÄ…trz metody klasy warstwy kontrolerĂłw, wywoĹ,ywane zostajÄ… operacje zdefiniowane w usĹ,ugach warstwy biznesowej. UsĹ,ugi te, realizowane mogÄ… byÄ‡ zarĂłwno wewnÄ…trz api jak i stanowiÄ‡ odrÄ™bny system internetowy. Klasy warstwy logiki biznesowej, zwane serwisami, zĹ,oĹĽone sÄ… z metod, ktĂłrych gĹ,Ăłwnym celem jest weryfikacja poprawnoĹ›ci otrzymanych informacji w kontekĹ›cie obsĹ,ugiwanych zasobĂłw, a takĹĽe pozyskiwanie danych oraz wykonywanie operacji na nich, poprzez odwoĹ,ywanie siÄ™ do metod warstwy dostÄ™pu do danych.

ZbiĂłr klas warstwy dostÄ™pu do danych, stanowi ostatni z logicznych poziomĂłw, definiowanych w ramach architektury API. Fragmenty kodu zdefiniowane w tej warstwie, zwane repozytoriami, majÄ… za zadanie obsĹ,uĹĽyÄ‡ komunikacjÄ™ pomiÄ™dzy interfejsem programowania aplikacji, a okreĹ›lonym ĹşrĂłdĹ,em danych. Ponadto, metody klas repozytoriĂłw, dostarczajÄ… warstwie logiki biznesowej interfejs operacji na danych. DziÄ™ki temu, ĹĽÄ…danie moĹĽe byÄ‡ przetwarzane od warstw najwyĹĽszych (tj. warstwy kontrolerĂłw) do warstwy najniĹĽszej (tj. warstwy dostÄ™pu do danych), natomiast odpowiedĹş na ĹĽÄ…danie jest konsolidowana w kierunku odwrotnym [**?**]. Na ilustracji **??** przedstawiono przepĹ,yw informacji wewnÄ…trz interfejsu API, od momentu wygenerowania ĹĽÄ…dania do chwili uzyskania odpowiedzi.



Rys. 2.1: Proces przetwarzania ŁŀÄ…dania wewnÄ…trz interfejsu API

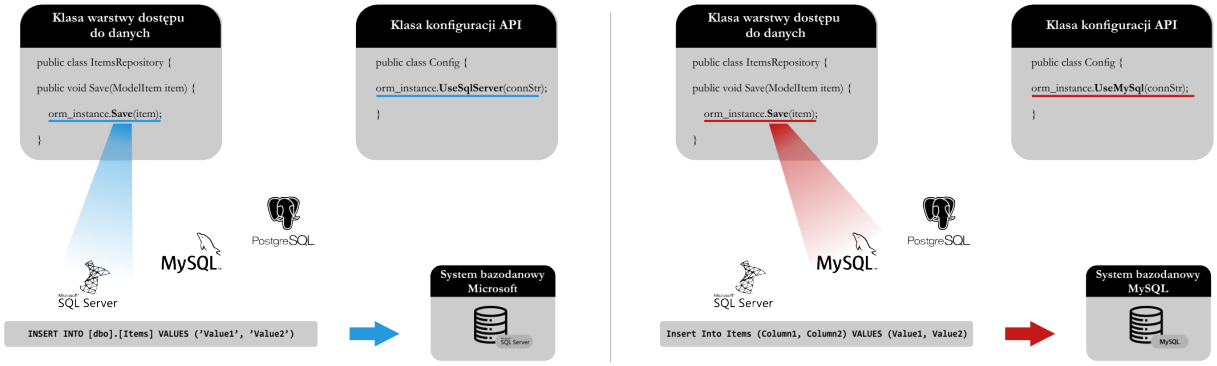
Konwersja obiektowo-relacyjna

W celu uproszczenia procesu pozyskiwania oraz modyfikacji danych z zewnÄ™trznych ŁşrĂłdeŁ,, a takŁŀe unifikacji sposobu interakcji z nimi, w ramach interfejsĂłw programowania aplikacji, powszechnie wykorzystywane jest oprogramowanie zwane mapperem obiektowo-relacyjnym (*ang. Object-Relational Mapper*). ZaŁ,oŁŀeniem oprogramowania tego, jest zdefiniowanie warstwy abstrakcji pomiÄ™dzy interfejsem programowania aplikacji a jÄ™zykiem programowania bÄ…dŁş zbiorem poleceŁ„, wykorzystywanym w ramach obsŁ,ugi ŁşrĂłdŁ,a danych.

Podstawowe skŁ,adowe oprogramowania typu ORM to jednolity interfejs operacji na zbiorze danych, klasy kontekstu bazodanowego, a takŁŀe metody obsŁ,ugi komunikacji z bazÄ… danych.

DziÄ™ki wprowadzeniu jednolitego interfejsu operacji na danych, niezaleŁŀnie od ŁşrĂłdŁ,a informacji z jakim komunikuje siÄ™ API, wydanie konkretnego polecenia do dowolnego systemu bazodanowego rĂłwnoznaczne jest z kaŁŀdorazowym wywoŁ,aniem funkcji o takiej samej sygnaturze. StosujÄ…c takie podejŁ›cie, konstruktor interfejsu programowania aplikacji nie staje siÄ™ uzaleŁŀniony od ŁşrĂłdŁ,a danych z ktĂłrym pracuje. Ponadto, istnieje moŁŀliwoŁ›Ä‡ zamiany lub poŁ,Ä…czenia dodatkowego systemu bazodanowego, a operacja ta, nie wpŁ,ywa w jakikolwiek sposĂłb na dziaŁ,anie interfejsu API. Niniejsza zaleŁŀnoŁ›Ä‡ zostaŁ,a zilustrowana na rysunku **??**

Dystynktywnym elementem oprogramowania mappera obiektowo-relacyjnego jest klasa kontekstu bazodanowego. Klasa ta, jest kontenerem struktur w ramach ktĂłrych wyrĂłŁŀniÄ‡ moŁŀemy zbiory elementĂłw modelu danych, a takŁŀe konfiguracjÄ™ poszczegĂłlnych ich wŁ,aŁ›ciwoŁ›ci. PodstawowÄ… ideÄ… omawianej konwersji dziedziny obiektowej do domeny relacyjnej jest zdefiniowanie zbioru klas, opisujÄ…cych wykorzystywane zasoby, a nastÄ™pnie odwzorowanie ich w relacyjnym modelu danych, obsŁ,ugiwanym przez wybrany system bazodanowy. Klasa kontekstu pozwala na okreŁ›lenie, ktĂłre spoŁ›rĂłd struktur danych zdefiniowanych w ramach API powinny zostaÄ‡ rzutowane na obiekty tabel generowanych w obrÄ™bie bazy danych. Ponadto, dla wŁ,aŁ›ciwoŁ›ci kaŁŀdej z klas modelu danych, zdefinio-



Rys. 2.2: Zasada dziaĹ‚ania oprogramowania mappera obiektowo-relacyjnego w kontekĹ›cie jednolitego interfejsu operacji na zbiorze danych

waÄ‡ naleĹĽy konfiguracjÄ™, ktÄ…ra zostanie przetransformowana do modelu relacyjnego. W zakresie klasy kontekstu bazy danych, opisywane sÄ… takĹĽe relacje, jakie majÄ… zostaÄ‡ wygenerowane pomiÄ™dzy poszczegÄ…lnymi elementami modelu.

W celu nawiÄ…zania, utrzymania, a takĹĽe zakoĹ„czenia komunikacji z zewnÄ™trznym ĹşrÄ…dĹ‚em danych, oprogramowanie ORM wykorzystuje klasy zwane konektorami. Klasy te, dostarczajÄ… przejrzysty interfejs obsĹ‚ugi poĹ‚Ä…czenia, ktÄ…ry nastÄ™pnie jest opakowywany w zunifikowany interfejs, dostÄ™pny bezpoĹ›rednio dla twÄ…rcy API [**?**].

Uwierzytelnienie oraz autoryzacja

Proces uwierzytelnienia oraz autoryzacji uĹĽytkownika odwoĹ‚ujÄ…cego siÄ™ do interfejsu programowania aplikacji, przedstawiÄ‡ naleĹĽy w trzech nastÄ™pujÄ…cych krokach.

Pierwszym z nich, jest wygenerowanie ĹĽÄ…dania odwoĹ‚ujÄ…cego siÄ™ do punktu koĹ„cowego odpowiedzialnego za obsĹ‚ugÄ™ uwierzytelnienia wewnÄ…trz API. Ĺ»Ä…danie to, musi posiadaÄ‡ ciaĹ‚o, zawierajÄ…ce informacje poĹ›wiadczajÄ…ce o konkretnymi uĹĽytkowniku. NajczÄ™Ĺ›ciej, informacjÄ… tÄ…, jest nazwa uĹĽytkownika oraz hasĹ‚o.

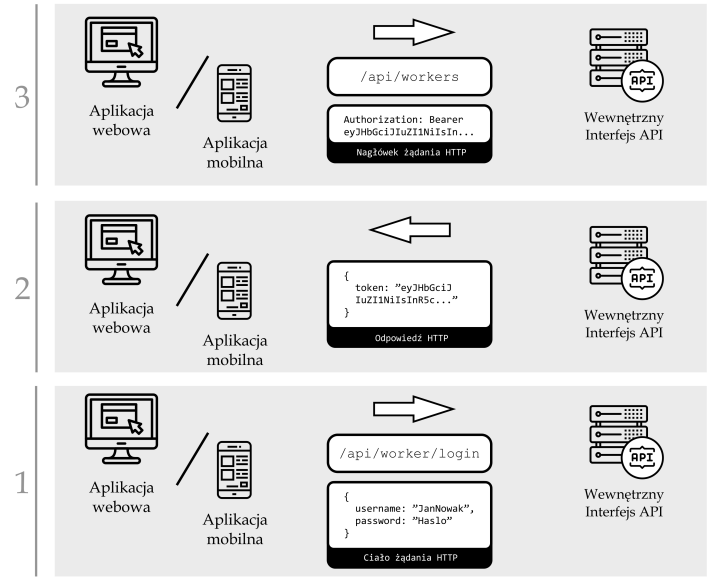
NastÄ™pnie, dostarczone referencje sÄ… analizowane przez mechanizmy uwierzytelniania implementowane w ramach API. W rezultacie tych operacji, zwrÄ…cona zostaje pozytywna odpowiedĹş zawierajÄ…ca token autoryzujÄ…cy bÄ…dĹş teĹĽ negatywna, posiadajÄ…ca w sobie informacjÄ™ o bĹ‚Ä™dzie uwierzytelnienia klienta.

Strona kliencka moĹĽe autoryzowaÄ‡ dysponowane operacje przed interfejsem programowania aplikacji, uwzglÄ™dniajÄ…c w ramach linii nagĹ‚Ä…wkowej ĹĽÄ…dania token uwierzytelniajÄ…cy. Dostarczona w ten sposÄ…b informacja, pozwala na identyfikacjÄ™ uĹĽytkownika w ramach interfejsu API, a takĹĽe na okreĹ›lenie przypisanego uĹĽytkownikowi poziomu uprawnieĹ„. W ramach struktury tokenu, zawarta jest takĹĽe informacja o jego czasie waĹĽnoĹ›ci, dlatego teĹĽ, procedura uwierzytelniania musi byÄ‡ regularnie ponawiana [**?**].

Na rysunku **??**, zilustrowany zostaĹ‚, proces uwierzytelnienia i autoryzacji aplikacji klienta przez interfejsem programowania aplikacji.

Separacja zapytaĹ„ oraz komend w kontekĹ›cie odwoĹ‚aĹ„ do ĹşrÄ…deĹ‚ danych

Wraz z rosnÄ…cÄ… liczbÄ… ĹĽÄ…daĹ„ obsĹ‚ugiwanych w ramach zaawansowanych interfejsÄ…w programowania aplikacji, zauwaĹĽone zostaĹ‚o zjawisko asymetrii w kontekĹ›cie typÄ…w wiadomoĹ›ci generowanych przez klientÄ…w. Zapytania dotyczÄ…ce pozyskiwania



Rys. 2.3: Proces uwierzytelnienia oraz autoryzacji uĹŽytkownika przed interfejsem API

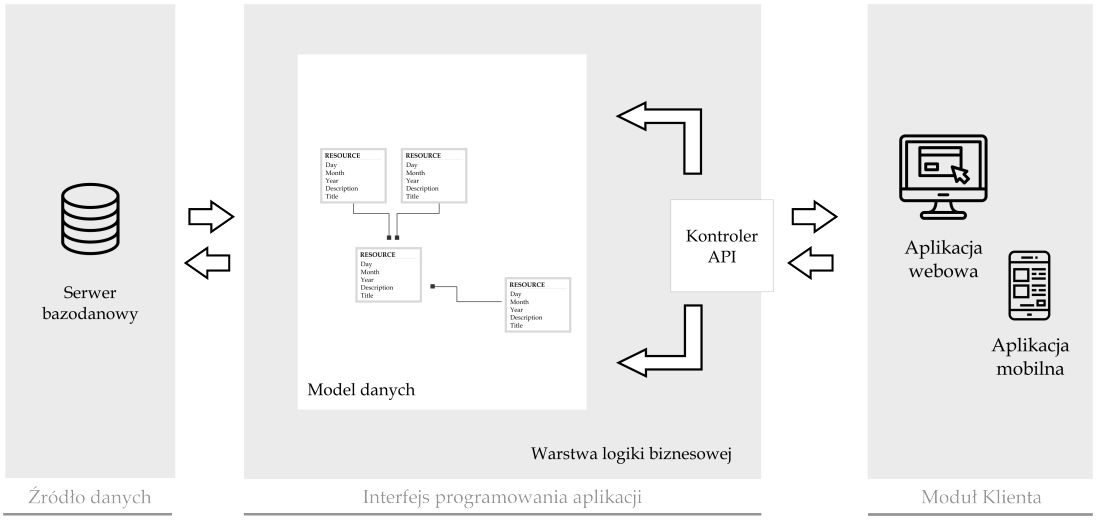
danych z API realizowane jest z nieporĂłwnywalnie wiÄ™kszÄ… czÄ™stoĹ›ciÄ… niĹż operacje ich modyfikacji. Dlatego teĹż, zdefiniowany zostaĹ, wzorzec projektowy dotyczÄ…cy separacji zapytaĹ„ oraz komend generowanych wzglÄ™dem usĹ‚ugi sieciowej *(ang. Command Query Responsibility Segregation)*.

Zastosowanie przedstawionego powyĹżej wzorca projektowego wiÄ…Ĺże siÄ™ z koniecznoĹ›ciÄ… budowy dwĂłch osobnych modeli danych. Pierwszy z nich, wykorzystywany jest w kontekĹ›cie odczytu informacji. Na modelu tym, dokonywana jest najczÄ™Ĺ›ciej operacja optymalizacji, ktĂłrej celem jest redukcja rozmiaru skĹ‚adowych modelu, a takĹże szybkoĹ›ci przetwarzania bardziej zĹ‚oĹĽonych struktur bÄ™dÄ…cych jego czÄ™Ĺ›ciÄ…. Drugi z modeli danych, znajduje zastosowanie w aspekcie modyfikacji okreĹ›lonych zasobĂłw. BiorÄ…c pod uwagÄ™ standardowy sposĂłb eksploatacji interfejsu programowania aplikacji, model ten cechowaÄ‡ siÄ™ moĹże niĹższÄ… wydajnoĹ›ciÄ…. W zaleĹżnoĹ›ci od specyfiki okreĹ›lonej usĹ‚ugi sieciowej, optymalizowany moĹże byÄ‡ albo model odczytu, albo teĹż model zapisu. Ponadto, implementujÄ…c wzorzec projektowy separacji zapytaĹ„ oraz komend, wykorzystaÄ‡ moĹżna dwa osobne zewnÄ™trzne ĹşrĂłdĹ‚a danych, przystosowane do wydajniejszego wykonywania okreĹ›lonego typu operacji, bÄ…dĹş teĹż dostosowane do obsĹ‚ugi wiÄ™kszego ruchu sieciowego.

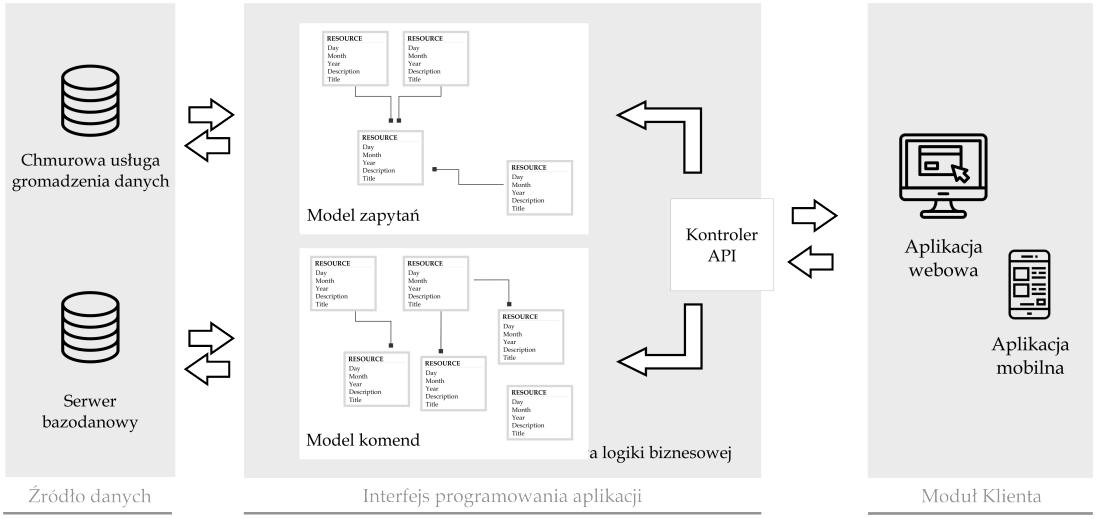
NiewÄ…tpliwymi zaletami, wynikajÄ…cymi z zastosowania opisywanego wzorca projektowego sÄ…: zwiÄ™kszenie efektywnoĹ›ci operacji realizowanych z duĹĽÄ… czÄ™stotliwoĹ›ciÄ…, moĹĽliwoĹ›Ä‡ korzystania z osobnych ĹşrĂłdeĹ‚, danych dla operacji odczytu oraz zapisu, zachowanie zasady pojedynczej odpowiedzialnoĹ›ci *(ang. Single Responsibility Principle)* wzglÄ™dem klas logiki biznesowej API, a takĹże redukcja liczby wstrzykiwanych zaleĹżnoĹ›ci *(ang. Dependency Injection)* w ramach klas kontrolerĂłw interfejsu [**?**].

Na ilustracjach **??** oraz **??** przedstawiono kolejno schemat przetwarzania ĹĽÄ…daĹ„ wewnÄ…trz API z uwzglÄ™dnieniem wzorca CQRS, a takĹże przy wykorzystaniu pojedynczego modelu danych.

OdwoĹ‚ujÄ…c siÄ™ do ilustracji **??** naleĹży zaznaczyÄ‡, Ĺże jest to jedna z moĹĽliwoĹ›ci implementacji wzorca CQRS, uwzglÄ™dniajÄ…ca wykorzystanie odseparowanych ĹşrĂłdeĹ‚, danych. Architektura separacji zapytaĹ„ oraz komend w kontekĹ›cie ĹĽÄ…daĹ„



Rys. 2.4: Schemat przetworzenia ĹŁÄ…dania przez interfejs API dla architektury z jednym modelem danych



Rys. 2.5: Schemat przetworzenia ĹŁÄ…dania przez interfejs API dla architektury wykorzystujÄ…cej wzorzec projektowy CQRS

wysyĹ‚anych do interfejsu API, moĹĽe byÄ‡ takĹĽe wprowadzona przy wykorzystaniu pojedynczego systemu bazodanowego.

Konwencja REST

Niezależnie od struktury wewnÄ™trznej omawianych usĹ‚ug sieciowych, wspĂłĹ‚czesne interfejsy programowania aplikacji projektowane sÄ… tak, aby ich zewnÄ™trzna warstwa (tj. widziana z perspektywy aplikacji klienckiej) cechowaĹ‚a siÄ™ jednolitÄ… kompozycjÄ….

Jednym z najpopularniejszych sposobĂłw zapewnienia jednolitego interfejsu komunikacyjnego pomiÄ™dzy klientami i usĹ‚ugami sieciowymi, przetwarzajÄ…cymi informacje z wykorzystaniem protokoĹ‚u HTTP, jest konwencja oraz styl architektoniczny REST *(ang. Representational State Transfer)*.

Konwencja ta, definiuje zbiĂłr zasad dotyczÄ…cych m.in. zachowania usĹ‚ugi sieciowej w kontekĹ›cie przetwarzania ĹŁÄ…dania konkretnego typu, struktury i elementĂłw odpowiedzi na okreĹ›lone ĹŁÄ…danie, semantyki wykorzystywanych statusĂłw rezultatu przetwarzania,

bezstanowego charakteru komunikacji, czy teĹĽ syntaktyki odwoĹ‚aĹ„ do poszczegĂłlnych punktĂłw koĹ„cowych.

W kontekĹ›cie stopnia implementacji stylu architektonicznego REST w ramach interfejsu programowania aplikacji, wprowadziÄ‡ naleĹĽy pojÄ™cie modelu dojrzaĹ‚oĹ›ci Richardsona *(ang. Richardson Maturity Model)*. PojÄ™cie to, definiuje cztery poziomy przystosowania interfejsu API do omawianej w niniejszej sekcji konwencji.

W odniesieniu do poziomu zerowego, powinnoĹ›ciÄ… interfejsu programowania aplikacji jest udostÄ™pnienie usĹ‚ug w ramach pojedynczego adresu sieciowego, niezaleĹĽnie od wykorzystywanych metod HTTP. Struktura ĹĽądania klienckiego, w sposĂłb jednoznaczny dostarczaÄ‡ ma informacjÄ™ na temat wykonywanego wewnÄ…trz usĹ‚ugi sieciowej dziaĹ‚ania.

Zasada poziomu pierwszego, odnosi siÄ™ do charakterystyki interfejsu API jako usĹ‚ugi zorientowanej na zasoby. NiezaleĹĽnie od czynnoĹ›ci, jaka ma zostaÄ‡ wykonana przez omawianÄ… usĹ‚ugÄ™ sieciowÄ…, opis tej czynnoĹ›ci wskazywaÄ‡ ma na zasĂłb ktĂłrego ona dotyczy.

ReguĹ‚a stanowiÄ…ca definicjÄ™ poziomu trzeciego, zwiÄ…zana jest z semantykÄ… poszczegĂłlnych typĂłw ĹĽÄ…daĹ„ protokoĹ‚u hipertekstowego. Ĺ»Ä…danie o takim samym adresie sieciowym, peĹ‚niÄ‡ powinno odmiennÄ… rolÄ™, w zaleĹĽnoĹ›ci od rodzaju ĹĽÄ…dania HTTP.

Ostatnim z poziomĂłw dojrzaĹ‚oĹ›ci interfejsu programowania aplikacji opartego o konwencjÄ™ REST jest reguĹ‚a HATEOAS *(ang. Hypertext As The Engine Of Application State)*. ReguĹ‚a ta, definiuje interfejs API jako ĹşrĂłdĹ‚o informacji dotyczÄ…cej obsĹ‚ugi stanu caĹ‚ego systemu internetowego (tj. usĹ‚ugi sieciowej wraz z aplikacjami klienckimi). Klient, po uzyskaniu odpowiedzi serwera na ĹĽÄ…danie, powinien na podstawie zawartoĹ›ci tej odpowiedzi mĂłc zdefiniowaÄ‡ przyszĹ‚e czynnoĹ›ci, ktĂłre wolno mu wykonaÄ‡ [**?**].

2.3. Testowanie oprogramowania

Aspekt badawczy niniejszej pracy, zwiÄ…zany jest z realizacjÄ… procesu ewaluacji wydajnoĹ›ci interfejsĂłw programowania aplikacji, pod kÄ…tem wykorzystania odmiennych Ĺ›rodowisk implementacyjnych oraz uruchomieniowych. Proces ten, jest tylko jednym z wielu elementĂłw domeny testowania oprogramowania, ktĂłrej charakterystyka uwzglÄ™dnia zbiĂłr sztywno zdefiniowanych reguĹ‚, cechujÄ…cych siÄ™ wysokim poziomem sformalizowania. W nastÄ™pnych sekcjach niniejszego akapitu dokonane zostaĹ‚o wprowadzenie dotyczÄ…ce zagadnienia ewaluacji oprogramowania, nakreĹ›lone zostaĹ‚y zasady testowania systemĂłw informatycznych, zdefiniono taksonomiÄ™ technik testowania, a takĹĽe omĂłwiono proces przeprowadzania ewaluacji wydajnoĹ›ci usĹ‚ugi sieciowej jakÄ… jest interfejs programowania aplikacji.

Wprowadzenie do zagadnienia ewaluacji oprogramowania

Ewaluacja poszczegĂłlnych skĹ‚adowych tworzonego oprogramowania jest niezbÄ™dnÄ… czÄ™Ĺ›ciÄ… procesu budowy systemu informatycznego,niezaleĹĽnie od jego charakterystyki, czy teĹĽ wykorzystywanej do jego budowy technologii. W rozumieniu ogĂłlnym, proces testowania czÄ™sto sprowadzany jest do zbioru dwĂłch czynnoĹ›ci. CzynnoĹ›ciami tymi sÄ… uruchamianie oprogramowania, a takĹĽe eksploracja jego funkcjonalnoĹ›ci w celu dostrzegania tych, w ramach ktĂłrych zauwaĹĽyÄ‡ moĹĽna niezgodnoĹ›Ä‡ ich dziaĹ‚ania w stosunku do specyfikacji. Takie wnioskowanie jednak jest niepeĹ‚ne, i uwzglÄ™dnia ono tylko jeden z etapĂłw skĹ‚adających siÄ™ na caĹ‚y proces testowania. DziedzinÄ™ ewaluacji cech programĂłw komputerowych, poszerzyÄ‡ naleĹĽy ponadto o takie elementy jak: plano-

wanie testĂłw, wybĂłr kryteriĂłw oceny oprogramowania, nadzĂłr oraz kontrolÄ™ realizacji badaĹ„, projektowanie przypadkĂłw testowych, czy teĹź analizÄ™ speĹ‚nienia ustalonych kryteriĂłw zakoĹ„czenia.

WyrĂłĹźniÄ‡ moĹźemy znaczÄ…cÄ… liczbÄ™ definicji testowania oprogramowania, a kaĹźda z nich wprowadza inny poziom szczegĂłĹ‚owoĹ›ci. Ponadto, wiele spoĹ›rĂłd formuĹ‚owanych pojÄ™Ä‡ nawiÄ…zuje do rĂłĹźnych aspektĂłw omawianego procesu. Zgodnie z jednÄ… z najbardziej generycznych definicji, wprowadzonÄ… przez Hetzla w publikacji [**?**], proces testowania oprogramowania okreĹ›liĂ‡ naleĹźy jako zbiĂłr wszystkich czynnoĹ›ci, ktĂłre nakierowane sÄ… na weryfikacjÄ™ atrybutĂłw i wĹ‚aĹ›ciwoĹ›ci programu, a takĹźe sprawdzenie tego, czy okreĹ›lony system speĹ‚nia zaĹ‚oĹźone wymagania. Definicja ta, wzglÄ™dem wielu innych popularnych sformuĹ‚owaĹ„ dotyczÄ…cych ewaluacji oprogramowania, uwzglÄ™dnia moĹźliwoĹ›Ä‡ zastosowania statycznych technik testowania. Ponadto, jej autor bierze pod uwagÄ™ fakt, Ĺźe w ramach procesu ewaluacji, oceniany powinien byĂ‡ kaĹźdy z artefaktĂłw tworzonych w ramach systemu, a nie tylko i wyĹ‚Ä…cznie kod ĹşrĂłdĹ‚owy programu.

Taksonomia technik testowania

Jako jedno z podstawowych kryteriĂłw podziaĹ‚u technik testowania oprogramowania, wskazaĂ‡ naleĹźy rodzaj czynnoĹ›ci wykonywanej przez stronÄ™ testujÄ…cÄ…, ktĂłrej realizacja prowadzi do uzyskania charakterystyki programu poddanego ewaluacji. WedĹ‚ug kryterium tego, wyrĂłĹźniÄ‡ naleĹźy statyczne oraz dynamiczne techniki testowania.

Pierwsze, spoĹ›rĂłd przytoczonych metod, opierajÄ… siÄ™ na analizie artefaktĂłw oprogramowania (takich jak m.in.: kod ĹşrĂłdĹ‚owy, specyfikacja, dokumentacja, czy teĹź lista wymagaĹ„) bez ich wykonywania. Jako praktyczne przykĹ‚ady przedstawionej techniki, zdefiniowaĂ‡ naleĹźy: generowanie metryk kodu ĹşrĂłdĹ‚owego programu, analizÄ™ przepĹ‚ywu sterowania, formalne dowodzenie poprawnoĹ›ci dziaĹ‚ania, a takĹźe interpretacjÄ™ grafĂłw wywoĹ‚aĹ„.

Metody dynamiczne natomiast, zwiÄ…zane sÄ… z weryfikacjÄ… wĹ‚aĹ›ciwoĹ›ci poszczegĂłlnych elementĂłw systemu informatycznego w trakcie jego wykonywania. Ten rodzaj testowania, nie uwzglÄ™dnia formalnych struktur liczbowych, jakimi sÄ… grafy przepĹ‚ywu sterowania, czy teĹź metryki kodu ĹşrĂłdĹ‚owego programu. Zorientowany jest on, na odbiĂłr systemu z perspektywy korzystajÄ…cego z niego klienta.

Innym z rozwaĹźanych kryteriĂłw podziaĹ‚u technik testowania jest ich umiejscowienie metody wzglÄ™dem okreĹ›lonego fragmentu procesu wytwĂłrczego. W nawiÄ…zaniu do tego aspektu, zdefiniowaĂ‡ naleĹźy pojÄ™cie poziomu testĂłw, ktĂłre jest powiÄ…zaniem sposobu ewaluacji oprogramowania z etapem jego realizacji. IstotÄ… rozrĂłĹźniania danych poziomĂłw testĂłw jest zaĹ‚oĹźenie rĂłĹźnorodnoĹ›ci celĂłw testowania, a takĹźe testowanych obiektĂłw, okreĹ›lanych w kontekĹ›cie kaĹźdej z warstw. W odniesieniu do najbardziej popularnych systematyk poziomĂłw testowania wyrĂłĹźniÄ‡ moĹźna nastÄ™pujÄ…ce elementy:

- testy jednostkowe (zwane takĹźe moduĹ‚owymi lub testami komponentĂłw)
- testy integracyjne
- testy systemowe
- testy akceptacyjne

Pierwszy z poziomĂłw, dotyczy znajdowania niezgodnoĹ›ci specyfikacyjnych w obrÄ™bie logicznie oddzielonych jednostek oprogramowania *(ang. Software Units)*. KaĹźda z jednostek, powinna byĂ‡ testowana w izolacji od pozostaĹ‚ych elementĂłw systemu. Ze wzglÄ™du na zĹ‚oĹźonoĹ›Ä‡ rozwiÄ…zaĹ„ informatycznych, a takĹźe statystycznie wy-

soki wspĂłŁ,czynnik wzajemnych zaleŁĽnoŁ›ci moduŁ,owych, warunek ten czÄ™sto nie moŁĽe zostaÄ‡ speŁ,niony. W takich sytuacjach, aby dostarczyÄ‡ zaleŁĽnoŁ›Ä‡ do testowanej jednostki, budowane sÄ… moduŁ,y zastÄ™pcze, imitujÄ…ce poprawne zachowanie okreŁ›lonego fragmentu programu *(ang. Mocks)*. Omawiany poziom testowania, czÄ™sto postrzegany jest jako jeden z etapĂłw procesu wytwĂłrczego, szczegĂłlnie w ramach takich technik jak rozwĂłj oprogramowania napÄ™dzany testowaniem *(ang. Test Driven Development)*.

Celem kolejnego z poziomĂłw testowania jest weryfikacja poprawnoŁ›ci wspĂłŁ,oddziaŁ,ywania indywidualnych komponentĂłw, a takŁĽe prawidŁ,owoŁ›ci funkcjonowania interfejsĂłw definiowanych pomiÄ™dzy nimi. PrzykŁ,adem wspĂłŁ,dziaŁ,ania jednostek oprogramowania moŁĽe byÄ‡ wspĂłŁ,praca interfejsu programowania aplikacji, bÄ™dÄ…cego systemem poddawanym analizie w ramach niniejszej pracy, a takŁĽe okreŁ›lonego silnika bazodanowego. W zaleŁĽnoŁ›ci od licznoŁ›ci weryfikowanych powiÄ…zaŁ„ pomiÄ™dzy poszczegĂłlnymi jednostkami oprogramowania, a takŁĽe w odniesieniu do liczby samych moduŁ,Ăłw bÄ™dÄ…cych czÄ™Ł›ciÄ… testowanego fragmentu systemu, wyrĂłŁĽniÄ‡ moŁĽemy testy maŁ,ej oraz duŁĽej skali integracji.

Na temat testĂłw systemowych, naleŁĽy mĂłwiÄ‡ wtedy, gdy wszystkie z elementĂłw rozwiÄ…zania informatycznego zostaŁ,y ze sobÄ… powiÄ…zane w sposĂłb spĂłjny. Celem testĂłw, realizowanych w ramach tego poziomu, jest weryfikacja wysokopoziomowej funkcjonalnoŁ›ci oprogramowania, a takŁĽe wykonywanie scenariuszy ewaluacji systemu z poziomu regularnego uŁĽytkownika *(ang. End-to-End testing)*.

Ostatnim z wymienionych poziomĂłw ewaluacji sÄ… testy akceptacyjne. Przedmiotem oceny w ramach tego rodzaju testĂłw jest gotowe rozwiÄ…zanie informatyczne w postaci komercyjnego produktu. Podmiot odpowiedzialny za realizacjÄ™ omawianych testĂłw przygotowuje listÄ™ kryteriĂłw akceptacji *(ang. acceptance criteria)*, a nastÄ™pnie na podstawie obsŁ,ugi testowanego rozwiÄ…zania, potwierdza lub odrzuca speŁ,nienie kaŁĽdego z nich. Celem omawianych ewaluacji nie jest znajdowanie bŁ,Ä™dĂłw dziaŁ,ania systemu, a nabranie zaufania co do jakoŁ›ci jego funkcjonalnych oraz niefunkcjonalnych atrybutĂłw.

WykonujÄ…c testy definiowane w ramach kolejnych poziomĂłw ewaluacji, weryfikowane zostajÄ… na poczÄ…tek funkcjonalne, a nastÄ™pnie niefunkcjonalne elementy testowanego systemu. Jako weryfikacjÄ™ elementĂłw funkcjonalnych, rozumieÄ‡ naleŁĽy wszystkie te czynnoŁ›ci, ktĂłre podejmowane sÄ… w ramach wszystkich wymienionych powyŁĽej poziomĂłw testĂłw, z wyjÄ…tkiem testĂłw akceptacyjnych. Ewaluacja niefunkcjonalna natomiast, odnosi siÄ™ tylko do ostatniego spoŁ›rĂłd wyrĂłŁĽnionych poziomĂłw testowania.

PodziaŁ, charakteryzujÄ…cy przedmiot ewaluacji wzglÄ™dem omĂłwionych aspektĂłw definiuje pojÄ™cia testĂłw funkcjonalnych oraz niefunkcjonalnych i w kontekŁ›cie niniejszej pracy jest on podziaŁ,em kluczowym.

Badania przeprowadzone w ramach tej pracy posiadajÄ… charakter testĂłw niefunkcjonalnych, a ich wykonanie poprzedzone jest weryfikacjÄ… funkcjonalnÄ…, ktĂłrej poprawnoŁ›Ä‡ traktowana jest jako wymĂłg.

Ewaluacja wydajnoŁ›ci interfejsĂłw programowania aplikacji

Zgodnie z teoriÄ… przedstawionÄ… w sekcji **??** niniejszej pracy interfejs programowania aplikacji postrzegaÄ‡ moŁĽna jako deterministyczny system wejŁ›ciowo-wyjŁ›ciowy o charakterze dyskretnym. Takie podejŁ›cie, w znaczÄ…cy sposĂłb uŁ,atwia proces ewaluacji wydajnoŁ›ci interfejsĂłw API.

Definiowanie interfejsu API jako systemu pobudzanego pojedynczym wejŁ›ciem, a takŁĽe generujÄ…cego pojedynczÄ… wartoŁ›Ä‡ wyjŁ›ciowÄ…, pozwala na wykorzystanie sposobu oceny wydajnoŁ›ci zwanego testem czarnoskrzynkowym *(ang. Black-box testing)*. W ramach tego rodzaju testu, okreŁ›lone kryterium ewaluacji wyliczane jest jako rĂłŁĽnica war-

toĹ›ci pomiaru na wyjĹ›ciu systemu, wzglÄ™dem tej, ktÄ…rej kalkulacja nastÄ…piĹ‚a na jego wejĹ›ciu. Taki rodzaj testu, umoĹĽliwia wyliczenie metryki wydajnoĹ›ci, bez koniecznoĹ›ci przygotowywania systemu do przeprowadzenia procesu testowania.

Podstawowym kryterium oceny wydajnoĹ›ci interfejsu programowania aplikacji jest czas odpowiedzi na ĹĽÄ…danie. MetrykÄ™ tÄ™, okreĹ›liÄ‡ naleĹĽy jako czas od momentu wygenerowania ĹĽÄ…dania przez stronÄ™ klienta, do chwili uzyskania przez niego odpowiedzi. ZauwaĹĽyÄ‡ naleĹĽy rÄ…wnieĹĽ zaleĹĽnoĹ›Ä‡ czasu odpowiedzi na ĹĽÄ…danie, zarÄ…wno od rozmiaru ĹĽÄ…dania jak i wielkoĹ›ci jego odpowiedzi. Ponadto, czynnikiem wpĹ‚ywajÄ…cym na uzyskany rezultat pomiaru jest niewÄ…tpliwie przepustowoĹ›Ä‡ Ĺ‚Ä…cza sieciowego pomiÄ™dzy klientem a serwerem.

Aby rezultaty uzyskane w ramach oceny wydajnoĹ›ci API mogĹ‚y zostaÄ‡ postrzegane jako rzetelne, omÄ…wione powyĹĽej czynniki muszÄ… cechowaÄ‡ siÄ™ statycznÄ… charakterystykÄ…, bÄ…dĹş teĹĽ zostaÄ‡ caĹ‚kowicie wyeliminowane z procesu testowego. W ramach niniejszej pracy, rozmiar odpowiedzi generowanej przez interfejsy programowania aplikacji jest staĹ‚y niezaleĹĽnie od zastosowanej technologii, a takĹĽe Ĺ›rodowiska uruchomieniowego. Wynika to z zastosowania pojedynczego i ustrukturyzowanego modelu danych, ktÄ…ry jest identyczny, niezaleĹĽnie od API. W odniesieniu do zmiennoĹ›ci prÄ™dkoĹ›ci Ĺ‚Ä…cza internetowego, aspekt ten zostaĹ‚, wyeliminowany poprzez przeprowadzanie testÄ…w w obrÄ™bie lokalnej sieci komputerowej, a takĹĽe umiejscowienie interfejsÄ…w oraz systemÄ…w bazodanowych w ramach tej wĹ‚aĹ›nie sieci.

Kolejnym z kryteriÄ…w oceny wydajnoĹ›ci, uwzglÄ™dnianym w ramach procesu testowania usĹ‚ug sieciowych jest poprawnoĹ›Ä‡ uzyskanej odpowiedzi w odniesieniu do liczby klientÄ…w, rÄ…wnolegle generujÄ…cych ĹĽÄ…dania. Kryterium to, charakteryzuje siÄ™ silnÄ… korelacjÄ… z czasem odpowiedzi na pojedyncze zapytanie.

UwzglÄ™dniajÄ…c oba opisane powyĹĽej parametry, podstawowy schemat scenariusza badawczego dotyczÄ…cego ewaluacji wydajnoĹ›ci API skĹ‚ada siÄ™ z nastÄ™pujÄ…cych testÄ…w:

- testy linii bazowej - pojedynczy klient generuje ĹĽądania w kierunku API w celu zdefiniowania Ĺ›redniego czasu odpowiedzi usĹ‚ug w standardowych warunkach jej dziaĹ‚ania. W ramach niniejszej pracy, podczas wykonywania omawianego testu, zdefiniowane zostanÄ… ponadto wartoĹ›ci wspÄ…ĹĽczynnikÄ…w satysfakcji, tolerancji oraz frustracji, bÄ™dÄ…ce skĹ‚adowymi wskaĹşnika jakoĹ›ci APDEX. WartoĹ›ci te, stanowiÄ… uogÄ…lnionÄ… ocenÄ™ wydajnoĹ›ci i posĹ‚uĹĽÄ… jako punkt odniesienia dla kolejnych testÄ…w.
- testy obciÄ…ĹĽeniowe - uwzglÄ™dniana zostaje zmienna liczba klientÄ…w generujÄ…cych ĹĽÄ…dania, w kontekĹ›cie ktÄ…rej ustalane sÄ… Ĺ›rednie czasu odpowiedzi na ĹĽÄ…danie, a takĹĽe dokonywane zostaje odniesienie uzyskanego rezultatu do wspÄ…ĹĽczynnikÄ…w zdefiniowanych uprzednio w ramach miary APDEX.
- testy przeciÄ…ĹĽajÄ…ce - liczba klientÄ…w generujÄ…cych ĹĽÄ…danie zostaje dobrana w taki sposÄ…b, aby doprowadziÄ‡ do obciÄ…ĹĽenia testowanej usĹ‚ugi, niepozwalajÄ…cego na poprawne funkcjonowanie interfejsu programowania aplikacji. Ewaluacja ta, ma na celu znalezienie punktu krytycznego w kontekĹ›cie dziaĹ‚ania testowanego oprogramowania.

Poza czarnoskrzynkowymi testami wydajnoĹ›ci, cechujÄ…cymi siÄ™ omÄ…wionÄ… powyĹĽej strukturÄ…, przeprowadzone mogÄ… zostaÄ‡ ewaluacje efektywnoĹ›ci dziaĹ‚ania poszczegĂłlnych fragmentÄ…w usĹ‚ugi sieciowej, w ramach ktĂłrych interfejs programowania aplikacji postrzegaÄ‡ naleĹĽy w odmienny sposÄ…b, aniĹĽeli jako system wejĹ›ciowo-wyjĹ›ciowy.

PrzykĹ‚adem oceny wydajnoĹ›ci okreĹ›lonego fragmentu interfejsu programowania aplikacji moĹĽe byÄ‡ ewaluacja moduĹ‚u realizacji zaawansowanych operacji obliczeniowych, do-

stÄ™pnego z poziomu punktu koŁ„cowego API. W takim przypadku, oprogramowanie musi zostaÄ‡ dostosowane do przeprowadzenia procedury testowej, a metryka wydajnoŁ›ci - powiÄ…zana z postÄ™pem realizacji obliczeŁ„.

2.4. Wykorzystywane narzÄ™dzia i technologie

ZarÃłwno w trakcie procesu implementacji badanych interfejsÃłw programowania aplikacji, jak i procedurze przeprowadzenia badaŁ„ pod kontem ich wydajnoŁ›ci, wykorzystano obszerny zbiÃłr sprawdzonych i powszechnie stosowanych rozwiÄ…zaŁ„ technologicznych. W ramach niniejszej sekcji, opisane zostanie kaŁĽde z nich.

C#

C# jest wieloparadygmatowym, a takŁĽe nowoczesnym jÄ™zykiem programowania ogÃłlnego przeznaczenia, charakteryzujÄ…cym siÄ™ bezpieczeŁ„stwem i niezawodnoŁ›ciÄ… w aspekcie typowania struktur danych. Pierwsza z wersji tego jÄ™zyka, stworzona zostaŁ,a przez Andersa Hejlsberga w roku 1998. Od tamtej chwili, do momentu napisania niniejszej pracy, upublicznionych zostaŁ,o 9 kolejnych, stabilnych wydaŁ„ projektu C#. KaŁĽda z nastÄ™pnych wersji omawianego jÄ™zyka programowania wprowadzaŁ,a zarÃłwno usprawnienia w kontekŁ›cie ekosystemu budowy i kompilacji programÃłw ŁşrÃłdŁ,owych, jak i wzbogacaŁ,a interfejs bibliotek funkcyjnych o kluczowe z punktu widzenia doŁ›wiadczonego programisty rozwiÄ…zania. Do rozwiÄ…zaŁ„ tych, zaliczyÄ‡ naleŁĽy miÄ™dzy innymi: mechanizmy programowania wspÃłŁ‚,bieŁĽnego, typy anonimowe, operatory zmiennych typÃłw niezdefiniowanych, obsŁ,ugÄ™ referencji, typy generyczne, czy teŁĽ wyraŁĽenia lambda.

W ramach niniejszej pracy, jÄ™zyk C# wykorzystany zostaŁ, do implementacji jednego z dwÃłch zbiorÃłw interfejsÃłw programowania aplikacji. Ze wzglÄ™du na zastosowanie rozwiÄ…zaŁ„ z zakresu przetwarzania wspÃłŁ‚,bieŁĽnego (tj. operacji asynchronicznych oraz wielowÄ…tkowych) udostÄ™pnianych przez omawiany jÄ™zyk programowania, interfejsy API realizowane w tej technologii mogÄ… obsŁ,ugiwaÄ‡ w sposÃłb rÃłwnolegŁ,y ŁĽÄ…dania pochodzÄ…ce od wielu klientÃłw, a takŁĽe utrzymywaÄ‡ sekwencyjny charakter przetwarzanych procedur niezależnie od czasu ich wykonywania. NaleŁĽy takŁĽe zwrÃłciÄ‡ uwagÄ™ na mechanizm wewnÄ™trznych usprawnieŁ„ wydajnoŁ›ciowych implementowany w ramach kompilatora i uruchamiany w momencie tŁ,umaczenia kodu jÄ™zyka do tzw. jÄ™zyka poŁ›redniego *(ang. Intermediate Language)*. DziÄ™ki zastosowaniu przedstawionego mechanizmu, operacje zdefiniowane przez programistÄ™ mogÄ… byÄ‡ modyfikowane w procesie kompilacji, tak aby nie wpŁ,ynÄ…Ä‡ na zaimplementowanÄ… funkcjonalnoŁ›Ä‡, zwiÄ™kszajÄ…c jednoczeŁ›nie wydajnoŁ›Ä‡ generowanego programu [**?**].

.NET Core

.NET Core postrzegaÄ‡ naleŁĽy jako Ł›rodowisko budowy, kompilacji oraz wykonywania rozwiÄ…zaŁ„ implementowanych w jÄ™zyku C#. Przedstawiana technologia stanowi podzbiÃłr bibliotek, dziÄ™ki ktÃłrym programista jest w stanie budowaÄ‡ systemy rÃłŁĽnorodnego przeznaczenia, a takŁĽe uruchamiaÄ‡ je w wielu wspieranych Ł›rodowiskach programowych. W przeciwieŁ„stwie do technologii .NET Framework bÄ™dÄ…cej poprzednikiem .NET Core, aplikacje tworzone na bazie omawianej biblioteki mogÄ… byÄ‡ wydawane nie tylko na system operacyjny Windows, ale takŁĽe na systemy Linux oraz MacOS.

W ramach omawianego Ł›rodowiska wykorzystywany zostaje komponent jÄ™zyka C# zwany bibliotekÄ… standardowÄ… *(ang. .NET Standard Library)*. Biblioteka ta jest

wspĂłlna dla wielu Ĺ›rodowisk uruchomieniowych, a zawarte w niej funkcjonalnoĹ›ci, traktowaÄ‡ naleĹĽy jako metody ogĂłlnego przeznaczenia.

Ponadto, Ĺ›rodowisko .NET Core, w ramach procesu budowy i kompilacji rozwiÄ…zania nawiÄ…zuje komunikacjÄ™ z komponentem wspĂłlnej infrastruktury *(ang. Common Infrastructure)*. Komponent ten, podobnie jak biblioteka standardowa, wspĂłĹ‚dzielony jest przez wiele Ĺ›rodowisk wykonawczych. W kontekĹ›cie wspĂłlnej infrastruktury, wspomnieÄ‡ naleĹĽy o wspĂłlnej specyfikacji jÄ™zyka *(ang. CLS - Common Language Specification)*, wspĂłlnym systemie typĂłw *(ang. CTS - Common Type System)*, a takĹĽe Ĺ›rodowisku uruchomieniowym wspĂłlnego jÄ™zyka *(ang. CLR - Common Language Runtime)*. Wykorzystanie miÄ™dzy innymi tych trzech elementĂłw, pozwala na budowÄ™ systemu dostÄ™pnego na wielu platformach [**?**].

W kontekĹ›cie realizowanej pracy, technologia .NET Core uĹĽyta zostaĹ‚a jako Ĺ›rodowisko uruchomieniowe dla interfejsĂłw programowania aplikacji tworzonych w jÄ™zyku C#. W obrÄ™bie technologii tej, poza przedstawionymi powyĹĽej komponentami, wyrĂłĹĽniÄ‡ moĹĽemy natywnÄ… bibliotekÄ™ ASP.NET Core, stanowiÄ…cÄ… zbiĂłr metod przydatnych w procesie definiowania internetowych usĹ‚ug sieciowych oraz aplikacji webowych. DziÄ™ki zastosowaniu ASP.NET Core operacje takie jak, miÄ™dzy innymi: obsĹ‚uga definicji kontrolerĂłw API, zarzÄ…dzanie stanem ciaĹ‚a ĹĽÄ…dania oraz jego rzutowaniem na okreĹ›lony typ danych, czy teĹĽ implementacja mechanizmĂłw uwierzytelniania i autoryzacji klienta, wykonane mogÄ… zostaÄ‡ na wysokim poziomie abstrakcji z jednoczesnym zapewnieniem naleĹĽytego poziomu ich wydajnoĹ›ci.

Entity Framework Core

Entity Framework Core stanowi narzÄ™dzie stworzone przez firmÄ™ Microsoft, ktĂłrego zastosowaniem jest mapowanie obiektowo-relacyjne realizowane w kontekĹ›cie usĹ‚ug internetowych tworzonych z wykorzystaniem jÄ™zyka C# oraz uruchamianych na platformie .NET Core. Przedstawiana biblioteka zapewnia programiĹ›cie zorientowany obiektowo interfejs, za pomocÄ… ktĂłrego moĹĽe on uzyskaÄ‡ dostÄ™p do danych, a takĹĽe je definiowaÄ‡ oraz przetwarzaÄ‡. Zbiory obiektĂłw mogÄ… byÄ‡ skĹ‚adowane zarĂłwno w relacyjnych jak i niereleacyjnych bazach danych. Niniejsza biblioteka, podobnie do Ĺ›rodowiska uruchomieniowego .NET Core, jest rozwiÄ…zaniem wieloplatformowym i moĹĽe byÄ‡ wykorzystywana przy budowie systemĂłw internetowych wdraĹĽanych na systemach Windows, Linux oraz MacOS.

Zastosowanie biblioteki mapera obiektowo-relacyjnego jakÄ… jest Entity Framework Core umoĹĽliwia zastosowanie podejĹ›cia zorientowanego na kod ĹşrĂłdĹ‚owy w kontekĹ›cie aplikacji komunikujÄ…cych siÄ™ i wykorzystujÄ…cych zewnÄ™trzne zbiory danych *(ang. Code-First Approach)*. PodejĹ›cie to, polega na definiowaniu w ramach kodu ĹşrĂłdĹ‚owego zbioru klas modelu danych, ktĂłre bÄ™dÄ… nastÄ™pnie przeksztaĹ‚cane do postaci tabel okreĹ›lonego systemu bazodanowego. Przedstawiona operacja przeksztaĹ‚cenia wykonywana jest bezpoĹ›rednio za pomocÄ… mechanizmĂłw mapera obiektowo-relacyjnego.

NiezaprzeczalnÄ… zaletÄ™ wykorzystania biblioteki ORM jakÄ… jest Entity Framework Core stanowi moĹĽliwoĹ›Ä‡ operowania na jednolitym interfejsie realizacji operacji na danych, niezaleĹĽnie od obsĹ‚ugiwanego systemu bazodanowego. Oznacza to, ĹĽe w momencie zmiany dostawcy zewnÄ™trznego ĹşrĂłdĹ‚a danych, zawartoĹ›Ä‡ kodu ĹşrĂłdĹ‚owego programu nie musi podlegaÄ‡ modyfikacji [**?**].

MediatR

MediatR to otwartoŚşrĂłdŚ‚owa biblioteka jÄ™zyka C#, z wykorzystaniem ktĂłrej zaimplementowany moŚźe zostaÄ‡ wzorzec projektowy, dotyczÄ…cy separacji odpowiedzialnoŚ›ci za obsŚ‚ugÄ™ zapytaŚ„ oraz komend przetwarzanych przez usŚ‚ugÄ™ sieciowÄ…. Kluczowym elementem biblioteki MediatR jest para generycznych interfejsĂłw, za pomocÄ… ktĂłrych implementowana jest obsŚ‚uga zarĂłwno ŚźÄ…daŚ„ jak i zapytaŚ„ dotyczÄ…cych danych. Interfejsami tymi sÄ… kolejno: *IRequest* - struktura programistyczna implementowana przez klasy definiujÄ…ce zawartoŚ›Ä‡ ciaŚ‚a ŚźÄ…dania lub komendy, a takŚźe powiÄ…zany z niÄ… *IRequestHandler*, ktĂłry jest konkretyzowany przez klasÄ™ definicji metody obsŚ‚ugi ŚźÄ…dania bÄ…dŚş operacji na danych.

Ponadto, naleŚźy rĂłwnieŚź podkreŚ›liÄ‡ znaczenie metody *Send* dostÄ™pnej w ramach gŚ‚Ăłwnego API pakietu MediatR. DziÄ™ki niej, wywoŚ‚ana moŚźe zostaÄ‡ procedura obsŚ‚ugi okreŚ›lonego ŚźÄ…dania lub operacji, z dowolnego miejsca kodu ŚşrĂłdŚ‚owego interfejsu programowania aplikacji [**?**].

JavaScript

JavaScript to wielofunkcyjny oraz wieloplatformowy skryptowy jÄ™zyk programowania cechujÄ…cy siÄ™ wysokim poziomem abstrakcji. Najbardziej popularnym przeznaczeniem omawianego jÄ™zyka jest budowa systemĂłw internetowych, a takŚźe mobilnych. HistorycznÄ… rolÄ… technologii JavaScript byŚ‚o udostÄ™pnianie programiŚ›cie funkcjonalnoŚ›ci umoŚźliwiajÄ…cych okreŚ›lanie rĂłŚźnorodnych sposobĂłw interakcji pomiÄ™dzy uŚźytkownikiem serwisu internetowego, a jego statycznymi elementami. Podstawowym Ś›rodowiskiem wykonania oraz interpretacji omawianego jÄ™zyka byŚ‚a uwczeŚ›nie przeglÄ…darka internetowa. Wraz z pojawieniem siÄ™ serwerowego Ś›rodowiska uruchomieniowego NodeJS, przeznaczonego dla jÄ™zyka JavaScript, popularnoŚ›Ä‡ omawianej technologii wzrosŚ‚a w gwaŚ‚townym tempie. Zmianie uległo rĂłwnieŚź gŚ‚Ăłwne przeznaczenie technologii, ktĂłra od tej pory staŚ‚a siÄ™ peŚ‚noprawnym jÄ™zykiem programowania, stosowanym w kontekŚ›cie budowy zarĂłwno systemĂłw internetowych, rozwiÄ…zaŚ„ mobilnych, jak i programĂłw desktopowych.

JÄ™zyk JavaScript uznaÄ‡ naleŚźy za technologiÄ™ charakteryzujÄ…cÄ… siÄ™ typowaniem sŚ‚abym oraz dynamicznym. W zwiÄ…zku z zastosowaniem przez twĂłrcĂłw rozwiÄ…zania takiego wŚ‚aŚ›nie podejŚ›cia, tworzone kody programĂłw naraŚźone sÄ… na wystÄ™powanie zjawisk niezgodnoŚ›ci typĂłw, a takŚźe niejawnej koercji. Ponadto, w kontekŚ›cie mechanizmĂłw omawianego jÄ™zyka, realizacja operacji przetwarzania wspĂłŚ‚bieŚźnego oraz wykonania metod asynchronicznych, zależna jest w caŚ‚kowitym stopniu od rozwiÄ…zaŚ„ implementacyjnych poczynionych w ramach Ś›rodowiska uruchomieniowego. Oznacza to, Ś›e przetwarzanie i wykonywanie operacji wielowÄ…tkowych moŚźe cechowaÄ‡ siÄ™ zrĂłŚźnicowanÄ… wydajnoŚ›ciÄ…, w zależnoŚ›ci od konkretnego interpretera jÄ™zyka.

NiewÄ…tpliwymi zaletami technologii JavaScript sÄ…: skŚ‚adnia cechujÄ…ca siÄ™ niskim poziomem zŚ‚oŚźonoŚ›ci poleceŚ„, moŚźliwoŚ›Ä‡ dowolnego wykorzystywania wielu spoŚ›rĂłd wspieranych paradygmatĂłw programowania, moduŚ‚owoŚ›Ä‡ i skalowalnoŚ›Ä‡ implementowanych rozwiÄ…zaŚ„, a takŚźe elastycznoŚ›Ä‡ w kontekŚ›cie operowania na wykorzystywanych strukturach danych [**?**].

W ramach niniejszej pracy, jÄ™zyk JavaScript zastosowany zostaŚ‚, w celu implementacji jednego z dwĂłch zbiorĂłw badanych interfejsĂłw programowania aplikacji. Tworzone w omawianym jÄ™zyku API, wykonywane bÄ™dÄ… w Ś›rodowisku uruchomieniowym NodeJS.

TypeScript

TypeScript stanowi statycznie typowany nadzbiÃłr jÄ™zyka JavaScript. OkreÅ›lenie to, oznacza Åże omawiana technologia nie jest stricte jÄ™zykiem programowania, a tylko okreÅ›lonÄ… grupÄ… instrukcji oraz procedur, ktÃłre wÅ‚Ä…czyÄ‡ moÅżna do jÄ™zyka JavaScript, po to, aby zapewniÄ‡ w jego kontekÅ›cie statyczny sposÃłb typowania danych. Technologia TypeScript nie moÅże byÄ‡ wykorzystywana samodzielnie, a Å›rodowisko wykonawcze JavaScript jest wymagane w celu uruchomienia skompilowanego moduÅ‚u, definiowanego zgodnie ze skÅ‚adniÄ… omawianego jÄ™zyka.

Kluczowym elementem przedstawianej technologii jest transpilator jÄ™zyka TypeScript o nazwie tsc *(ang. TypeScript Compiler)*. Program ten, uruchamiany jest tuÅż przed rozpoczÄ™ciem procedury interepretacji kodu JavaScript i przeksztaÅ‚ca on metody odpowiedzialne za obsÅ‚ugÄ™ typÃłw danych, do struktur dostÄ™pnych w ramach standardowej implementacji jÄ™zyka. Dlatego teÅż, z punktu widzenia Å›rodowiska uruchomieniowego, dostÄ™pne programistom mechanizmy definicji typÃłw czy interfejsÃłw, nie sÄ… znane.

Celem zastosowania omawianego nadzbioru jÄ™zykowego jest moÅżliwoÅ›Ä‡ kontroli zgodnoÅ›ci definiowanych obiektÃłw programistycznych pod kÄ…tem ich wewnÄ™trznej struktury. Ponadto, wykorzystanie TypeScript umoÅżliwia weryfikacjÄ™ faktu nieumyÅ›lnego odwoÅ‚ania siÄ™ do struktury typu nieokreÅ›lonego, jeszcze przed rozpoczÄ™ciem procesu interpretacji kodu [**?**].

NodeJS

NodeJS jest Å›rodowiskiem uruchomieniowym jÄ™zyka JavaScript zbudowanym w oparciu o otwartoÅşrÃłdÅ‚owy silnik interpretacji kodu Chrome V8. DziÄ™ki zastosowaniu omawianego Å›rodowiska uruchomieniowego, kod ÅşrÃłdÅ‚owy jÄ™zyka JavaScript moÅże byÄ‡ wykonywany poza ekosystemem przeglÄ…darki internetowej. RozwÃłj niniejszej technologii, doprowadziÅ‚, do diametralnej zmiany w obszarze zastosowania jÄ™zyka JavaScript, a takÅże gwaÅ‚townego wzrostu jego popularnoÅ›ci w kontekÅ›cie budowy systemÃłw internetowych.

Podobnie do rozwiÄ…zania Microsoft .NET Core, platforma NodeJS skÅ‚ada siÄ™ nie tylko ze Å›rodowiska uruchomieniowego, ale takÅże ze zbioru bibliotek oraz narzÄ™dzi linii komend. Aplikacje budowane na bazie omawianej technologii cechujÄ… siÄ™ zastosowaniem architektury sterowanej zdarzeniami *(ang. Event-driven architecture)*, ktÃłra ponadto wzbogacana jest (dziÄ™ki wykorzystaniu mechanizmu pÄ™tli) o moÅżliwoÅ›Ä‡ obsÅ‚ugi operacji asynchronicznych. Co wiÄ™cej, rozwiÄ…zania definiowane na podstawie Å›rodowiska NodeJS posiadajÄ… budowÄ™ moduÅ‚owÄ…, co przyczynia siÄ™ do zwiÄ™kszenia ich zdolnoÅ›ci w kontekÅ›cie skalowania systemÃłw.

NaleÅży rÃłwnieÅż uwypukliÄ‡ generycznoÅ›Ä‡ charakterystyki Å›rodowiska NodeJS. Nie jest ono przeznaczone stricte do definiowania i wdraÅżania usÅ‚ug sieciowych, a wykorzystywane jest do uruchamiania dowolnego kodu jÄ™zyka JavaScript, jaki moÅże zostaÄ‡ stworzony za pomocÄ… jego skÅ‚adni. Dlatego teÅż, aby dostarczaÄ‡ mechanizmy dotyczÄ…ce specyficznych funkcjonalnoÅ›ci, ekosystem NodeJS moÅże byÄ‡ rozbudowywany poprzez otwartoÅşrÃłdÅ‚owe moduÅ‚y. LicznoÅ›Ä‡ moduÅ‚Ãłw tych, a takÅże popularnoÅ›Ä‡ ich wykorzystania stanowi niewÄ…tpliwie o sile omawianej technologii [**?**].

ExpressJS

ExpressJS stanowi bibliotekÄ™ Å›rodowiska NodeJS, dostarczajÄ…cÄ… zbiÃłr metod pozwalajÄ…cych na budowÄ™ webowych interfejsÃłw programowania aplikacji w tym wÅ‚aÅ›nie Å›rodowisku. Pakiet ExpressJS cechuje siÄ™ minimalizmem w kon-

tekŁcie zŁoŁonoŁci udostÄ™pnianych programiŁcie operacji, elastycznoŁciÄ… dotyczÄ…cÄ… wspÓŁpracy z zewnÄ™trznymi pakietami, a takŻe wysokÄ… wydajnoŁciÄ… dziaŁania tworzonych aplikacji, poprzez eliminacjÄ™ zŁoŁonych funkcjonalnoŁci przetwarzania zasobÓw w obrÄ™bie ŻÄ…dania.

Koncepcja przetwarzania zasobu uzyskanego od klienta w ramach ExpressJS sprowadza siÄ™ do potokowej obsŁugi dostarczonego wejŁcia, przez kolejne funkcje poŁredniczÄ…ce *(ang. Middleware functions)*. Ostatnia z funkcji ma za zadanie zwrÓciÄ‡ odpowiedŹ wygenerowanÄ… na podstawie operacji wykonywanych przez wszystkie poprzednie metody. CiaŁem funkcji poŁredniczÄ…cej moŻe byÄ‡ zarÓwno kod ŹródŁowy zdefiniowany przez programistÄ™, jak i ten dostarczony poprzez referencjÄ™ do zewnÄ™trznego pakietu. Do kaŻdej z metod przekazywane sÄ… parametry Ż Ä…dania, odpowiedzi, a takŻe referencji do nastÄ™pnego middleware. DziÄ™ki uruchomieniu napisanej w taki sposÓb aplikacji w Łrodowisku NodeJS, poszczegÓlne funkcje poŁredniczÄ…ce mogÄ… byÄ‡ realizowane asynchronicznie [**?**].

Prisma

Prisma ORM to narzÄ™dzie peŁniÄ…ce rolÄ™ mapera obiektowo-relacyjnego wykorzystywanego w kontekŁcie implementowania interfejsÓw programowania aplikacji w jÄ™zyku JavaScript. DystynktywnÄ… cechÄ… omawianego narzÄ™dzia jest prostota definiowania rzutowanego modelu danych. W przypadku pakietu Prisma, caŁa struktura modelu danych opisywana jest w ramach jednego pliku zwanego plikiem schematu. W pliku tym, okreŁlane zostajÄ… zarÓwno wŁaŁciwoŁci poszczegÓlnych encji modelu, jak i relacje ktÓre miÄ™dzy nimi wystÄ™pujÄ….

Analogicznie do narzÄ™dzia Entity Framework Core, Prisma ORM wspiera zarÓwno relacyjne jak i nierelacyjne systemy bazodanowe. Ponadto, zauwaŻyÄ‡ naleŻy kompatybilnoŁÄ‡ omawianego narzÄ™dzia z omawianÄ… wyŻej technologiÄ… TypeScript [**?**].

Mongoose

Biblioteka Mongoose stanowi narzÄ™dzie rzutowania obiektowego modelu danych definiowanego w ramach kodu programu, do postaci obiektowej nierelacyjnej bazy danych MongoDB. Technologii tej nie naleŻy nazywaÄ‡ maperem obiektowo-relacyjnym, gdyŻ wykonywane przez niÄ… operacje synchronizujÄ… struktury danych, ktÓre zarÓwno po stronie kodu, jak i po stronie ŹródŁa danych cechujÄ… siÄ™ obiektowÄ… naturÄ….

PoŁÄ…czenie interfejsu programowania aplikacji z nierelacyjnym ŹródŁem danych obsŁugiwanym poprzez bibliotekÄ™ Mongoose, prowadziÄ‡ moŻe do zwiÄ™kszenia efektywnoŁci oraz zmniejszenia czasu wykonania operacji na danych. JednakŻe, w zwiÄ…zku z naturÄ… przechowywanych informacji, a takŻe brakiem uwzglÄ™dniania w ich strukturze metadefinicji, zastosowanie omawianego mechanizmu moŻe rÓwnieŻ prowadziÄ‡ do braku spÓjnoŁci ŹródŁa danych, a takŻe niemoŻliwoŁci zastosowania usprawnieŁ„ wydajnoŁciowych w kontekŁcie bazy danych [**?**].

Apache JMeter

NarzÄ™dzie Apache JMeter to otwartoŹródŁowe oprogramowanie stworzone w jÄ™zyku Java. Program ten, wykorzystywany jest przeprowadzania ewaluacji wydajnoŁci oprogramowania sieciowego opierajÄ…cego swoje dziaŁanie o protokoŁy HTTP oraz FTP. Testy efektywnoŁci dziaŁania usŁug mogÄ… zostaÄ‡ przeprowadzane w trybie lokalnym (tj. z wykorzystaniem jednego hosta wysyŁajÄ…cego Ż Ä…dania do okreŁlonej usŁugi sieciowej),

jak i w trybie rozproszonym (tj. budowana zostaje hierarchia hostÃłw bÄ™dÄ…cych generatorami Ł'Ä…daÅ„). W ramach niniejszej pracy, zastosowany zostaÅ, drugi z przedstawionych trybÃłw ewaluacji.

DziaÅ,anie oprogramowania Apache JMeter sprowadza siÄ™ do wykonywania pÄ™tli testowej, w ramach okreÅ›lonych grup wÄ…tkÃłw. PÄ™tla testowa symuluje sekwencyjne generowanie Ł'Ä…daÅ„ w kierunku serwera, z uwzglÄ™dnieniem staÅ,ej wartoÅ›ci opÃłÅşnienia pomiÄ™dzy wysyÅ,anymi pakietami. Grupa wÄ…tkÃłw natomiast, okreÅ›la wspÃłÅ,bieÅ'ny charakter testÃłw obciÄ…Å'enia usÅ,ugi sieciowej i moÅ'e byÄ‡ utoÅ'samiana zarÃłwno z konkretnymi wÄ…tkami procesora lokalnego hosta, jak i z oddzielnie pracujÄ…cymi generatorami Ł'Ä…daÅ„ w trybie rozproszonym.

W celu zdefiniowania testu wydajnoÅ›ci w ramach Apache JMeter, zbudowany powinien zostaÄ‡ plan testowy. Jest to podstawowa jednostka wyrÃłÅ'niana w ramach niniejszego oprogramowania i skupia ona w sobie miÄ™dzy innymi komponenty grup wÄ…tkÃłw, prÃłbnikÃłw *(ang. Samplers)*, a takÅ'e elementÃłw nasÅ,uchujÄ…cych na odpowiedÅş usÅ,ugi *(ang. Listeners)*. ZarÃłwno prÃłbniki, jak i elementy nasÅ,uchujÄ…ce mogÄ… byÄ‡ powielane w ramach planu testowego, a takÅ'e indywidualnie konfigurowane, w zaleÅ'noÅ›ci od specyfiki usÅ,ugi sieciowej.

Oprogramowanie Apache JMeter obsÅ,ugiwaÄ‡ moÅ'na zarÃłwno z poziomu narzÄ™dzia linii komend, jak i udostÄ™pnionego graficznego interfejsu uÅ'ytkownika [**?**].

2.5. PrzeglÄ…d literatury

W niniejszym rozdziale przedstawione zostanÄ… pozycje literaturowe, do ktÃłrych odnosiÄ‡ siÄ™ bÄ™dzie opisywana praca dyplomowa. Pozycje te, podzielone zostaÅ,y na oddzielne grupy, zwiÄ…zane z okreÅ›lonÄ… tematykÄ….

Na poczÄ…tku, przedstawiona zostanie literatura powiÄ…zana z aspektem budowy interfejsÃłw programowania aplikacji oraz bÄ™dÄ…ca wprowadzeniem do wykorzystywanych technologii. NastÄ™pnie, opisane zostanÄ… pozycje traktujÄ…ce o wydajnoÅ›ci interfejsÃłw API, a takÅ'e o analizie dziaÅ,ania powszechnie dostÄ™pnych serwisÃłw internetowych opartych o metodologiÄ™ REST. Kolejne prace, skupiaÄ‡ siÄ™ bÄ™dÄ… na tematyce testowania usÅ,ug sieciowych, teorii testowania, a takÅ'e konfiguracji narzÄ™dzi dla testÃłw rozproszonych. W nastÄ™pnej kolejnoÅ›ci, wspomniane zostanÄ… prace naukowe oraz dokumenty standaryzacyjne dotyczÄ…ce sposobu dziaÅ,ania protokoÅ,u przesyÅ,ania danych hipertekstowych. OstatniÄ… grupÄ… pozycji literaturowych bÄ™dÄ… prace referencyjne dotyczÄ…ce badaÅ„ wydajnoÅ›ci systemÃłw internetowych.

Pozycja [**?**] stanowi wprowadzenie do zaawansowanych konceptÃłw jÄ™zyka C#, a takÅ'e dostarcza informacji zwiÄ…zanych z wykorzystaniem tego jÄ™zyka w Å›rodowiskach uruchomieniowych .NET oraz .NET Core. W poczÄ…tkowych rozdziaÅ,ach przedstawiono sposÃłb budowy, kompilacji oraz wykonywania programu w Å›rodowisku .NET. Kolejno opisana zostaÅ,a struktura bazowych aplikacji uruchamianych w tym wÅ,aÅ›nie Å›rodowisku i tworzonych za pomocÄ… jÄ™zyka C#. Ostatnim elementem wprowadzenia do opisywanej technologii byÅ,o przedstawienie struktur jÄ™zyka w kontekÅ›cie obiektowego paradygmatu programowania. W nastÄ™pnych sekcjach literatury, w sposÃłb wyczerpujÄ…cy poruszono tematykÄ™ bardziej zaawansowanych aspektÃłw programowania w jÄ™zyku C#, ktÃłrymi sÄ… miÄ™dzy innymi: kolekcje i typy generyczne, delegaty i wyraÅ'enia lambda, czy teÅ' cykl Å'ycia obiektu w pamiÄ™ci programu. WaÅ'nym tematem, poruszonym w ramach tej ksiÄ…Å'ki jest struktura oraz zasada dziaÅ,ania Å›rodowiska .net core, bÄ™dÄ…cego podstawowym elementem interfejsÃłw programowania aplikacji tworzonych w jÄ™zyku C#.

AnalogicznÄ… do przedstawionej powyÅĽej pozycji literaturowej, dotyczÄ…cÄ… jednak technologii NodeJS oraz jÄ™zyka JavaScript jest [**?**]. W ramach tej pracy zawarto obszerne wprowadzenie do platformy NodeJS uwzglÄ™dniajÄ…ce ponadto kwestie obsÅ‚ugi operacji wejÅ›cia/wyjÅ›cia, wykonywania natywnego kodu JS, czy teÅĽ przetwarzania operacji przez silnik NodeJS oraz bibliotekÄ™ libuv. Znaczna czÄ™Å›Ä‡ pracy, obejmuje przedstawienie zaawansowanych wzorcÃłw projektowych, ktÃłrych gÅ‚Ãłwnym przeznaczeniem jest obsÅ‚uga zdarzeÅ„ oraz operacji asynchronicznych. Wspomniane zostaÅ‚y takÅĽe rozwiÄ…zania dotyczÄ…ce skalowalnoÅ›ci aplikacji z wykorzystaniem mechanizmÃłw kolejkowania wiadomoÅ›ci.

NiezaleÅĽnie od wykorzystywanej technologii, interfejsy programowania aplikacji, ktÃłre zostaÅ‚y zbudowane na potrzeby tej pracy dyplomowej, oparte sÄ… o styl architektoniczny RESTful. Styl ten, jest pewnym zbiorem zasad projektowania usÅ‚ug sieciowych, okreÅ›lających zarÃłwno aspekty sposobu komunikacji klienta z usÅ‚ugÄ… sieciowÄ…, jak i techniczne wymagania dotyczÄ…ce przetwarzania ÅĽÄ…daÅ„. Dobre praktyki, ktÃłre uwzglÄ™dnia metodologia REST, zawarte zostaÅ‚y w pozycji literaturowej [**?**]. Autorzy tego dokumentu, na wstÄ™pie dokonujÄ… porÃłwnania architektury zorientowanej na zasoby, bÄ™dÄ…cej podstawÄ… konwencji REST, z popularnÄ… uprzednio architekturÄ… zorientowanÄ… na usÅ‚ugi. NastÄ™pnie, przedstawiane sÄ… najlepsze praktyki, cele oraz reguÅ‚y REST dotyczÄ…ce projektowania interfejsu programowania aplikacji. Co wiÄ™cej, w omawianej ksiÄ…ÅĽce zawarte zostaÅ‚y takÅĽe podstawowe oraz zaawansowane wzorce projektowania API, uwzglÄ™dniajÄ…ce aspekty bezstanowoÅ›ci, paginacji, osiÄ…galnoÅ›ci, a takÅĽe identyfikacji zasobÃłw interfejsu. KoÅ„cowe rozdziaÅ‚y ksiÄ…ÅĽki, wprowadzajÄ… w kwestie testowania oraz bezpieczeÅ„stwa REST API, omawiajÄ… technikÄ™ kompozycji usÅ‚ug RESTful, a takÅĽe przedstawiajÄ… rozwiÄ…zania (biblioteki oraz jÄ™zyki programowania) pozwalające na tworzenie interfejsÃłw API zgodnych z metodologiÄ… REST.

Podstawowym celem dziaÅ‚ania interfejsu programowania aplikacji jest dostarczenie danych do konsumenta, bÄ…dÅş teÅĽ ich manipulacja zgodnie z jego ÅĽÄ…daniem. Aby operowaÄ‡ na danych, interfejs API musi komunikowaÄ‡ siÄ™ ze ÅşrÃłdÅ‚em danych, ktÃłrym najczÄ™Å›ciej jest serwer bazodanowy. W celu dostarczenia metod komunikacji pomiÄ™dzy API a ÅşrÃłdÅ‚em danych, ktÃłre jednoczeÅ›nie sÄ… niezaleÅĽne od wykorzystywanego ÅşrÃłdÅ‚a, a takÅĽe pozwalają na zarzÄ…dzanie danymi z poziomu struktur jÄ™zyka, stworzone zostaÅ‚y biblioteki zwane maperami obiektowo-relacyjnymi (ang. Object-Relational Mappers). Dla API napisanego w jÄ™zyku C# podstawowym rozwiÄ…zaniem ORM jest biblioteka Entity Framework Core, ktÃłra przedstawiona zostaÅ‚a w pozycji [**?**]. Pozycja ta, uwzglÄ™dnia zarÃłwno opis działania najczÄ™Å›ciej wykorzystywanych metod sÅ‚uÅĽÄ…cych do manipulacji danymi, jak i rolÄ™ klasy kontekstu bazodanowego w procesie tÅ‚umaczenia operacji programistycznych na polecenia bazodanowe. Ponadto, dowiedzieÄ‡ moÅĽemy siÄ™ jak przetwarzaÄ‡ zaawansowane typy danych (takie jak np. DateTime), czy teÅĽ w jaki sposÃłb wykorzystywaÄ‡ zapytania LINQ do budowania kwerend.

Dla interfejsu programowania aplikacji napisanego w jÄ™zyku JavaScript i uruchamianego w Å›rodowisku NodeJS, w przeciwieÅ„stwie do platformy .NET, zastosowaÄ‡ moÅĽemy zdecydowanie wiÄ™kszÄ… liczbÄ™ bibliotek peÅ‚niących rolÄ™ maperÃłw obiektowo-relacyjnych. Biblioteki te, zostaÅ‚y opisane w pozycjach [**?**] i [**?**]. Pozycja [**?**] peÅ‚ni rolÄ™ caÅ‚oÅ›ciowego wprowadzenia do tematyki tworzenia interfejsÃłw API, korzystając z platformy NodeJS, frameworka ExpressJS oraz nierelacyjnej bazy danych MongoDB. RozdziaÅ‚ piÄ…ty tej pracy, traktujący o wykorzystaniu baz danych NoSQL, przybliÅĽa tematykÄ™ jednego z najczÄ™Å›ciej wykorzystywanych maperÃłw obiektowo-relacyjnych dla Node czyli mongoose. Przedstawiono tutaj sposÃłb zestawienia poÅ‚Ä…czenia z serwerem bazodanowym, tworzenia encji modelu, przekształcanego nastÄ™pnie na struktury bazy danych, a takÅĽe wykonywania operacji dostÄ™pu do danych i ich modyfikacji. W pracy [**?**] natomiast, po-

rÄłwnano nierelacyjne podejĹ›cie do skĹ‚adowania danych typu geograficznego z podejĹ›ciem relacyjnym, wykorzystujÄ…c w tym przypadku biblioteki mongoose i sequelize. Oba mapery obiektowo relacyjne zostaĹ‚y uĹĽyte w ramach interfejsu API wykorzystujÄ…cego technologie NodeJS/ExpressJS. Celem opisywanej pracy byĹ‚o przedstawienie rÄŁĹĽnic w czasach odpowiedzi API na uzyskane ĹĽÄ…danie, dla rÄŁĹĽnej liczby danych geolokalizacyjnych, uwzglÄ™dniajÄ…c zastosowanie relacyjnych i nierelacyjnych baz danych.

NastÄ™pne pozycje literaturowe, zwiÄ…zane sÄ… z analizÄ… usĹ‚ug REST oraz wydajnoĹ›ciÄ… webowych interfejsÓŁw programowania aplikacji.

Pozycja [**?**] stanowi analizÄ™ 500 serwisÓŁw internetowych z listy alexa.com4000 najpopularniejszych dostÄ™pnych publicznie usĹ‚ug sieciowych. TwÓŁrcy kaĹĽdego z 500 serwisÓŁw deklarujÄ… zgodnoĹ›Ä‡ swoich produktÓŁw z konwencjÄ… REST. Przeprowadzona analiza dotyczyĹ‚a kluczowych aspektÓŁw technicznych zwiÄ…zanych z funkcjonowaniem API, stopnia zgodnoĹ›ci API z reguĹ‚ami dotyczÄ…cymi metodologii REST, a takĹĽe przestrzegania najlepszych praktyk projektowania interfejsÓŁw programowania aplikacji, takich jak m.in. zastosowanie mechanizmu wersjonowania. W trakcie analizy, zaobserwowano okreĹ›lone trendy dla aplikacji REST API, takie jak m.in. rozpowszechnione wsparcie notacji JSON, czy wykorzystywanie narzÄ™dzi do dokumentacji generowanej programowo. Ponadto, zauwaĹĽono, ĹĽe tylko ok. 0.8% analizowanych serwisÓŁw webowych przestrzega w sposÓŁb Ĺ›cisĹ‚y reguĹ‚, zawartych w ramach konwencji REST.

WydajnoĹ›Ä‡ interfejsÓŁw programowania aplikacji, jako jeden z elementÓŁw miary jakoĹ›ci API zostaĹ‚a przedstawiona w pozycji [**?**]. Na poczÄ…tku pracy, jej autorzy wskazujÄ… na interakcjÄ™ interfejsu programowania aplikacji z systemami klienckimi. Opisany zostaĹ‚, tutaj zestaw protokoĹ‚Ół sieciowych wykorzystywanych podczas formuĹ‚owania i transmisji ĹĽÄ…dania, system zunifikowanych lokacji zasobÓŁw, a takĹĽe semantyka interakcji w zaleĹĽnoĹ›ci od wykorzystywanych typÓŁw ĹĽÄ…daĹ„ protokoĹ‚u hipertekstowego. Ponadto, wskazano najczÄ™stsze przyczyny bĹ‚Ä™dÓŁw przepĹ‚ywu danych dla http, uwzglÄ™dniajÄ…c dziaĹ‚anie usĹ‚ugi DNS, bĹ‚Ä™dy poĹ‚Ä…czenia, bĹ‚Ä™dy leĹĽÄ…ce po stronie klienta, a takĹĽe bĹ‚Ä™dy wynikajÄ…ce z dziaĹ‚ania serwera. Kolejna czÄ™Ĺ›Ä‡ pracy, zwiÄ…zana jest ze skĹ‚adowymi metryki jakoĹ›ci, do ktÓŁrych wedĹ‚ug autorÓŁw, poza wydajnoĹ›ciÄ…, zaliczyÄ‡ moĹĽemy: dostÄ™pnoĹ›Ä‡, procent ĹĽÄ…daĹ„ dla ktÓŁrych uzyskano pozytywnÄ… odpowiedĹş, osiÄ…galnoĹ›Ä‡, a takĹĽe moĹĽliwoĹ›Ä‡ sprawdzenia stanu usĹ‚ugi w dowolnym momencie jej dziaĹ‚ania. Dodatkowo, w niniejszej pracy zaproponowano podejĹ›cie oraz zestaw narzÄ™dzi pozwalajÄ…cych na dokonanie ewaluacji jakoĹ›ci interfejsu programowania aplikacji, zgodnie z przyjÄ™tÄ… normÄ… jakoĹ›ci.

Kolejnym etapem nastÄ™pujÄ…cym po zdefiniowaniu metryki wydajnoĹ›ci, jest ustalenie wartoĹ›ci tejĹĽe metryki w kontekĹ›cie testowanych usĹ‚ug sieciowych. Przytoczone poniĹĽej pozycje literaturowe, zwiÄ…zane sÄ… z wykonywaniem pomiarÓŁw wydajnoĹ›ci API, czyli testowaniem.

Pozycja [**?**] stanowi obszerne wprowadzenie do teorii testowania oprogramowania. W pierwszych rozdziaĹ‚ach tego dokumentu, wyjaĹ›niono czym jest testowanie, dlaczego jest ono niezbÄ™dne podczas tworzenia oprogramowania, a takĹĽe jak wyglÄ…da podstawowy proces wykonywania testÓŁw. NastÄ™pnie przedstawiono proces testowania w kontekĹ›cie tworzenia oprogramowania. UwzglÄ™dniono tu zarÓŁwno modele cyklu ĹĽycia rozwoju systemÓŁw w powiÄ…zaniu z testowaniem, poziomy realizowanych testÓŁw, ich typy, jak i sposoby zarzÄ…dzania testami. Kolejne rozdziaĹ‚y tyczÄ… siÄ™ testowania statycznego (tj. testowania funkcjonalnoĹ›ci lub moduĹ‚u na poziomie jego specyfikacji lub implementacji bez wykonywania kodu testowanego oprogramowania), dostarczajÄ… teorii zwiÄ…zanej z poszczegÓŁlnymi technikami testowania rozwiÄ…zaĹ„ oraz przedstawiajÄ… aspekt organizacji, planowania, monitorowania oraz uwzglÄ™dniania ryzyka w czasie dokonywania ewaluacji systemÓŁw. W

ostatnim z rozdziaÅ‚ów dokumentu, autorzy przedstawiajÄ… narzÄ™dzia przydatne w procesie testowania, a takÅ¼e sposÃ³b ich efektywnego wykorzystania w codziennej pracy.

Pozycja [**?**] zawiera wiele analogicznych treÅ›ci do pracy opisanej powyÅ¼ej, jednakÅ¼e rozwija ona w sposÃ³b wyczerpujÄ…cy, wspomniane tylko w poprzedniej pracy aspekty. W czÄ™Å›ci drugiej dokumentu zawarto dogÅ‚Ä™bnÄ… analizÄ™ zagadnienia testowania statycznego, uwzglÄ™dniajÄ…c m.in. testowanie zgodnoÅ›ci ze standardami oprogramowania, symboliczne wykonywanie kodu, a nawet wprowadzajÄ…c aparat matematyczny do formalnego dowodzenia poprawnoÅ›ci fragmentÃ³w oprogramowania. W ramach tej ksiÄ…Å¼ki, przedstawiono takÅ¼e dynamicznÄ… analizÄ™ systemu (tj. testowanie funkcjonalnoÅ›ci lub moduÅ‚u na poziomie wykonywanego kodu) uwzglÄ™dniajÄ…c czÄ™sto wystÄ™pujÄ…ce bÅ‚Ä™dy zwiÄ…zane m.in. z nieumiejÄ™tnym zarzÄ…dzaniem strukturami pamiÄ™ci programu. Ponadto, uwzglÄ™dniono zagadnienie priorytetyzacji przypadkÃ³w testowych, wprowadzajÄ…c pojÄ™cie miary Å›redniego procenta wykrytych usterek. Autor dokumentu przedstawia takÅ¼e testowanie charakterystyk jakoÅ›ciowych zgodnie z normÄ… ISO 9126 oraz ISO 25010, tworzenie dokumentacji w ramach zarzÄ…dzania testowaniem, czy chociaÅ¼by zarzÄ…dzanie incydentami wystÄ™pujÄ…cymi w ramach procesu ewaluacji oprogramowania.

W ramach pozycji [**?**], dowiedzieÄ‡ moÅ¼emy siÄ™ ponadto o testowaniu usÅ‚ug internetowych. Przedstawiono tutaj podstawowÄ… strukturÄ™ standardowej usÅ‚ugi sieciowej (w tym przypadku â€“ usÅ‚ugi e-commerce) cechujÄ…cej siÄ™ architekturÄ… trÃ³jwarstwowÄ…. Ponadto, wyjaÅ›niono rolÄ™ kaÅ¼dej z warstw systemu, a takÅ¼e przedstawiono aspekty testowania oprogramowania w kontekÅ›cie kaÅ¼dej z nich. Dodatkowo, zawarte zostaÅ‚y przykÅ‚adowe przypadki testowe, dotyczÄ…ce zarÃ³wno prezentacji danych w systemie, jak i dostÄ™pu do danych poprzez serwer webowy. Dla zaprezentowanych przypadkÃ³w testowych, przedstawione zostaÅ‚y takÅ¼e scenariusze realizacji testÃ³w w postaci listy czynnoÅ›ci jakie naleÅ¼y podjÄ…Ä‡, aby dokonaÄ‡ ewaluacji systemu.

Aspekty technologii testowania oprogramowania ujÄ™te zostaÅ‚y takÅ¼e w pozycji [**?**]. ArtykuÅ‚ ten, stanowi sekcjÄ™ wprowadzajÄ…cÄ… do ksiÄ…Å¼ki pt. Tutorial: Software Testing and Valdation Techniques, tego samego autora. Pozycja ta, przedstawia przekrÃ³j technik oraz technologii testowania oprogramowania wykorzystywanych na przestrzeni ostatnich ok. 30 lat. Opisane zostaÅ‚y tutaj zarÃ³wno teoretyczne podstawy testowania, narzÄ™dzia i techniki analizy statycznej i dynamicznej, oceny efektywnoÅ›ci przeprowadzanych testÃ³w, a takÅ¼e badania przeprowadzane w dziedzinie testowania i walidacji oprogramowania. Omawiany artykuÅ‚, wyszczegÃ³lnia pozytywne oraz negatywne aspekty poszczegÃ³lnych technik oraz wskazuje przydatnoÅ›Ä‡ okreÅ›lonych rozwiÄ…zaÅ„ do testowania oprogramowania rÃ³Å¼nego typu.

OstatniÄ… przytoczonÄ… w ramach tego przeglÄ…du literaturowego pozycjÄ…, dotyczÄ…cÄ… teorii ewaluacji oprogramowania jest [**?**]. Pozycja ta, stanowi normÄ™ miÄ™dzynarodowej organizacji normalizacyjnej (ang. International Organization for Standardization) dotyczÄ…cÄ… weryfikacji jakoÅ›ci oprogramowania. UwzglÄ™dniono tu przede wszystkim znaczenie pojÄ™Ä‡ stosowanych w dziedzinie testowania oprogramowania, wprowadzono definicje dla okreÅ›lonych terminÃ³w oraz zjawisk wystÄ™pujÄ…cych w ramach ewaluacji systemÃ³w, a takÅ¼e okreÅ›lono zgodnoÅ›Ä‡ wprowadzanych przez standard konceptÃ³w, z konceptami zawartymi w standardach pochodnych. GÅ‚Ã³wnÄ… czÄ™Å›Ä‡ dokumentu, stanowi wprowadzenie szkieletu modelu jakoÅ›ci, uwzglÄ™dniającego okreÅ›lone modele jakoÅ›ciowe, modele jakoÅ›ci w uÅ¼yciu, a takÅ¼e modele jakoÅ›ci produktu. Dodatkowo, przedstawiono cel oraz sposÃ³b wykorzystania modeli jakoÅ›ciowych, wyjaÅ›niono rÃ³Å¼nicÄ™ w postrzeganiu modeli jakoÅ›ciowych z punktu widzenia rÃ³Å¼nych interesariuszy, a takÅ¼e zdefiniowano relacje pomiÄ™dzy okreÅ›lonymi modelami. Dokument

ten, wraz z normÄ… ISO 9126, stanowiÄ… definicjÄ™ pojÄ™cia jakoĹ›ci w kontekĹ›cie testowania oprogramowania.

Pozycja [**?**] stanowi przeglÄ…d narzÄ™dzi wykorzystywanych do testowania dziaĹ‚ania systemĂłw komputerowych. Na poczÄ…tku ksiÄ…ĹĽki, wprowadzany jest termin zapewnienia jakoĹ›ci (ang. Quality Assurance), ktĂłry w dzisiejszych czasach definiuje zakres odpowiedzialnoĹ›ci osoby testujÄ…cej oprogramowanie. Kolejno, przedstawiane sÄ… kryteria sukcesu dotyczÄ…ce tworzonego systemu, a takĹĽe fazy poszczegĂłlnych modeli rozwoju oprogramowania zorientowanych na procesy. Analogicznie do pozycji literaturowych przedstawionych uprzednio, w ramach tej pozycji okreĹ›lone zostaĹ‚y metryki i definicje jakoĹ›ci oprogramowania oraz omĂłwiony zostaĹ‚, proces realizacji testĂłw. GĹ‚Ăłwna czÄ™Ĺ›Ä‡ omawianego dokumentu skupiona jest wokĂłĹ‚, narzÄ™dzi stosowanych do realizacji ewaluacji oprogramowania. WyszczegĂłlniono tutaj narzÄ™dzie WinRunner, przedstawiajÄ…c miÄ™dzy innymi wykorzystywany w tym programie skryptowy jÄ™zyk testĂłw (ang. Test Script Language). Ponadto, przedstawiono architekturÄ™ oraz najwaĹĽniejsze funkcjonalnoĹ›ci narzÄ™dzi SilkTest, SQA Robot, LoadRunner, TestDirector, QuickTest Professional a takĹĽe Apache JMeter. Ostatni z wymienionych programĂłw, wykorzystywany zostanie w ramach niniejszej pracy dyplomowej, dlatego teĹĽ dalszy przeglÄ…d tej pozycji literaturowej skupiony bÄ™dzie na rozdziale dotyczÄ…cym wĹ‚aĹ›nie tego narzÄ™dzia. Opis funkcjonalnoĹ›ci aplikacji JMeter zostaĹ‚, w niniejszej pozycji podzielony na sekcje zwiÄ…zane z testowaniem rozwiÄ…zaĹ„, bazodanowych wykorzystujÄ…cych interfejs JDBC (ang. Java DataBase Connectivity), a takĹĽe sekcjÄ™ dotyczÄ…cÄ… testowania aplikacji bazujÄ…cych w swoim dziaĹ‚aniu na protokole hipertekstowym. Przedstawiono tutaj sposĂłb tworzenia grup wÄ…tkĂłw reprezentujÄ…cych uĹĽytkownikĂłw aplikacji, generowania ĹĽÄ…dania protokoĹ‚u hipertekstowego, uruchomienia mechanizmu nasĹ‚uchiwania na odpowiedĹş serwisu, dodawania licznika czasu, a takĹĽe zapisywania i przeglÄ…dania rezultatĂłw przeprowadzonego testu.

W ramach dokumentĂłw [**?**] oraz [**?**] przedstawiono peĹ‚en zakres funkcjonalnoĹ›ci dostÄ™pnych w ramach narzÄ™dzia Apache JMeter. Pierwsza z prac (tj. [**?**]), skupia siÄ™ na wykorzystaniu narzÄ™dzia w celu wykonywania testĂłw wydajnoĹ›ci usĹ‚ug sieciowych, natomiast druga z pozycji (tj. [**?**]), przedstawia aplikacjÄ™ JMeter dla rĂłĹĽnych kontekstĂłw jej potencjalnego uĹĽycia. W obu pracach wyszczegĂłlnione zostajÄ… podstawowe elementy, na ktĂłre skĹ‚ada siÄ™ Ĺ›rodowisko testowe. Elementami tymi sÄ…: grupy wÄ…tkĂłw, komponenty prĂłbkujÄ…ce, kontrolery, komponenty nasĹ‚uchujÄ…ce, liczniki czasu oraz asercje. Ponadto, omĂłwiono elementy graficznego interfejsu uĹĽytkownika dla aplikacji, przedstawiono proces instalacji oraz uruchamiania narzÄ™dzia JMeter, a takĹĽe zdefiniowano pojÄ™cie planu testĂłw. W kontekĹ›cie pracy [**?**], poza wymienionymi uprzednio kwestiami, zobrazowany zostaĹ‚, takĹĽe proces wykonywania testu przeciÄ…ĹĽeniowego dla usĹ‚ugi zorientowanej na serwisy (ang. Service-Oriented Application). Proces ten uwzglÄ™dniaĹ‚: tworzenie grupy wÄ…tkĂłw, konfiguracjÄ™ struktury ĹĽÄ…dania wysyĹ‚anego do usĹ‚ugi, uruchomienie testu, a takĹĽe pozyskanie wyniku. W pracy [**?**] natomiast, analogiczny proces, moĹĽemy zaobserwowaÄ‡ dla monolitycznej aplikacji internetowej oraz interfejsu programowania aplikacji. Ponadto, przedstawione zostaĹ‚y zaawansowane opcje konfiguracji elementĂłw nasĹ‚uchujÄ…cych oraz licznikĂłw czasu, a takĹĽe pokazany zostaĹ‚, proces wykorzystania poĹ›redniczÄ…cego serwera http, w celu dokumentowania realizowanych ĹĽÄ…daĹ„.

NastÄ™pne pozycje literaturowe omĂłwione w ramach tej pracy, dotyczÄ… budowy oraz zasady dziaĹ‚ania internetowego protokoĹ‚u hipertekstowego (ang. Hypertext Transfer Protocol), a takĹĽe implementacji mechanizmu zarzÄ…dzania stanem. Mechanizm ten, w zwiÄ…zku z naturÄ… protokoĹ‚u http, nie jest w nim domyĹ›lnie realizowany.

Pozycja [**?**] stanowi techniczny dokument dotyczÄ…cy semantyki oraz budowy internetowego protokoĹ‚u hipertekstowego w wersji 1.1. Zdefiniowano w nim pojÄ™cie zasobu

ÅĽÄ…dania oraz omÃłwiono cykl Åťycia jego przetwarzania. Wskazano takÅĽe moment, w ktÃłrym zasÃłb rekonstruowany jest przez serwer na podstawie jego efektywnego identyfikatora URI (ang. Uniform Resource Identifier). Ponadto, nakreÅ›lono pojÄ™cie reprezentacji danych przesyÅ‚anych za pomocÄ… protokoÅ‚u http, definiujÄ…c okreÅ›lone pola nagÅ‚Ã³wkowe dotyczÄ…ce: typu danych, sposobu kodowania, jÄ™zyka danych, a takÅĽe lokalizacji zasobu. Kolejne rozdziaÅ‚y dokumentu zawierajÄ… informacje dotyczÄ…ce definicji dozwolonych metod protokoÅ‚u http oraz znaczenia jakie te metody wprowadzajÄ… w kontekÅ›cie operacji na zasobie. W dokumencie przedstawiono takÅĽe kody statusu odpowiedzi na ÅĽÄ…dnie, grupujÄ…c je w sposÃłb semantyczny. Dla kaÅĽdego z przedstawionych kodÃłw statusu nakreÅ›lono kontekst, w jakim odpowiedÅş, oznaczona tym wÅ‚aÅ›nie kodem, powinna byÄ‡ zwracana klientowi. Na koÅ„cu pracy, omÃłwiono kwestie zwiÄ…zane z bezpieczeÅ„stwem protokoÅ‚u takie jak: ataki bazujÄ…ce na wstrzykiwaniu kodu czy ochrona przed ujawnianiem informacji wraÅĽliwych w identyfikatorach zasobÃłw.

Pozycja [**?**] pozwala na poszerzenie wiedzy dotyczÄ…cej protokoÅ‚u hipertekstowego w bardziej praktycznym kontekÅ›cie. Podobnie jak w dokumencie [**?**], przedstawiono tutaj informacje teoretyczne dotyczÄ…ce architektury protokoÅ‚u, definicji zasobÃłw czy teÅĽ ujednoliconego formatu ich adresowania. Ponadto, wskazano i scharakteryzowano okreÅ›lone typy poÅ‚Ä…czeÅ„ realizowanych z wykorzystaniem protokoÅ‚u hipertekstowego. Co wiÄ™cej, dla kaÅĽdego z nich rozwaÅĽono kwestie zwiÄ…zane z wydajnoÅ›ciÄ… poÅ‚Ä…czenia pomiÄ™dzy klientem a serwerem. Kolejne rozdziaÅ‚y pracy [**?**] traktujÄ… o identyfikacji klienta w ramach serwera, jego uwierzytelniania przed serwerem, a takÅĽe szyfrowania danych przesyÅ‚anych pomiÄ™dzy tymi dwiema jednostkami. W niniejszej pracy wspomniano takÅĽe o internacjonalizacji ÅĽÄ…daÅ„ w kontekÅ›cie zastosowania nagÅ‚Ã³wka â€Accept-Languageâ€™. Ostatnie rozdziaÅ‚y dokumentu dotyczÄ… kwestii publikowania i dystrybucji zawartoÅ›ci. WyszczegÃłlnione zostaÅ‚y tu takie elementy jak: web hosting, systemy publikacji treÅ›ci, czy teÅĽ mechanizm przekierowaÅ„ oraz rÃłwnowaÅĽenia obciÄ…ÅĽeÅ„.

Zgodnie z charakterystykÄ… protokoÅ‚u http, realizuje on komunikacjÄ™ w sposÃłb bezstanowy. Oznacza to, ÅĽe domyÅ›lnie, pomiÄ™dzy klientem a serwerem nie jest utrzymywana sesja poÅ‚Ä…czeniowa, a kaÅĽde ÅĽÄ…danie generowane przez klienta w kierunku serwera rozpatrywane jest indywidualnie. RozwiÄ…zanie takie, pozwala na znaczÄ…ce przyspieszenie dziaÅ‚ania protokoÅ‚u hipertekstowego, a takÅĽe uproszczenie jego konstrukcji. JednakÅĽe, szczegÃłlnie w przypadku aplikacji internetowych komunikujÄ…cych siÄ™ z serwerem http, bezstanowy charakter protokoÅ‚u bywa problematyczny w aspekcie kontekstu wysyÅ‚anych sekwencyjnie ÅĽÄ…daÅ„. Dlatego teÅĽ, do protokoÅ‚u http wprowadzono mechanizm zarzÄ…dzania stanem opisany w dokumencie [**?**]. Dokument ten, definiuje pola nagÅ‚Ã³wkowe o nazwach â€HTTP Cookieâ€™ oraz â€Set-Cookieâ€™. Pola te, mogÄ… byÄ‡ uÅĽywane przez serwery http w celu przechowywania stanu w ramach aplikacji klienckich, dajÄ…c serwerom tym moÅĽliwoÅ›Ä‡ zarzÄ…dzania, zawierajÄ…cÄ… stan sesjÄ…, przy wykorzystaniu protokoÅ‚u bezstanowego. W niniejszym dokumencie, dla obu przedstawionych pÄłl wyszczegÃłlniono atrybuty skÅ‚adowe pola, a takÅĽe okreÅ›lono znaczenie kaÅĽdego z nich. Ponadto, dokument definiuje wymagania dla klienta http, dotyczÄ…ce moÅĽliwoÅ›ci wykorzystania mechanizmu zarzÄ…dzania stanem. Pod uwagÄ™ wziÄ™te zostaÅ‚y takÅĽe kwestie bezpieczeÅ„stwa takie jak identyfikatory sesji, sÅ‚aba poufnoÅ›Ä‡ danych, czy teÅĽ zaufanie do usÅ‚ugi nazw domenowych w celu prawidÅ‚owego dziaÅ‚ania mechanizmu zarzÄ…dzania stanem.

Ostatnia grupa pozycji literaturowych, zawartych w ramach niniejszego przeglÄ…du literaturowego dotyczy badaÅ„ zwiÄ…zanych z testowaniem wydajnoÅ›ci aplikacji internetowych w Å›rodowisku rozproszonym. Pozycje przedstawione poniÅĽej, bÄ™dÄ… stanowiÄ‡ prace referencyjne wzglÄ™dem niniejszej pracy dyplomowej.

ArtykuÅ‚ [?] dotyczy porÃ³wnania wydajnoÅ›ci dziaÅ‚ania interfejsÃ³w programowania aplikacji tworzonych z wykorzystaniem platform .NET Core 3.1 oraz .NET 5. Celem powstania tego dokumentu byÅ‚a weryfikacja zjawiska wzrostu wydajnoÅ›ci dziaÅ‚ania programÃ³w, tworzonych i uruchamianych z wykorzystaniem nowszej z platform firmy Microsoft. Praca ta, ma takÅ¼e na celu pomÃ³c pozwoliÄ‡ odpowiedzieÄ‡ na pytanie, czy kod ÅşrÃ³dÅ‚owy interfejsu programowania aplikacji o okreÅ›lonych funkcjonalnoÅ›ciach, a takÅ¼e korzystajÄ…cy z okreÅ›lonych narzÄ™dzi, powinien zostaÄ‡ zaktualizowany w taki sposÃ³b, aby wspieraÄ‡ najnowszÄ…, stabilnÄ… wersjÄ™ Å›rodowiska .NET. W ramach dokumentu, w celu realizowania pomiarÃ³w wydajnoÅ›ci wykorzystano opisane w poprzednich akapitach narzÄ™dzie Apache JMeter, a takÅ¼e dedykowanÄ… Å›rodowisku .NET, bibliotekÄ™ BenchmarkDotNet. Kolejne rozdziaÅ‚y artykuÅ‚u przedstawiajÄ… przygotowane Å›rodowisko testowe, plan wykonywanych testÃ³w, a takÅ¼e uzyskane rezultaty wraz z ich analizÄ…. Autor pracy, zobrazowaÅ‚, wyniki szeÅ›ciu testÃ³w wydajnoÅ›ciowych, biorÄ…cych pod uwagÄ™ proces serializacji oraz deserializacji obiektÃ³w typu JSON za pomocÄ… bibliotek NewtonsoftJson, a takÅ¼e System.Text.Json. Ponadto, przygotowany zostaÅ‚, test wyszukiwania wzorca z obszernym ciÄ…gu tekstowym oraz test wykorzystania punktu koÅ„cowego jako klienta zewnÄ™trznego API. Na podstawie otrzymanych rezultatÃ³w, wnioskowaÄ‡ moÅ¼emy o okoÅ‚o 24 procentowym Å›rednim wzroÅ›cie wydajnoÅ›ci wykonywania operacji realizowanych w ramach testÃ³w. Ponadto, wykazano takÅ¼e doÅ›Ä‡ znaczÄ…cy (okoÅ‚o 35 procentowy) Å›redni spadek wydajnoÅ›ci nowego rozwiÄ…zania wzglÄ™dem poprzednika, w kontekÅ›cie testÃ³w obciÄ…Å¼eniowych.

Analogiczne badania przeprowadzono w ramach pracy [?]. W tym przypadku jednak, nie skupiaÅ‚y siÄ™ one na aspekcie porÃ³wnania technologii, a na sposobie wykonywania pomiarÃ³w, a takÅ¼e definiowaniu kryteriÃ³w oceny jakoÅ›ci. W pracy tej, interfejs programowania aplikacji zbudowany w oparciu o metodologiÄ™ REST poddawany byÅ‚, zmiennym obciÄ…Å¼eniom (tj. testy linii bazowej, testy obciÄ…Å¼eniowe oraz testy przeciÄ…Å¼eniowe). W czasie dokonywania ewaluacji monitorowano Å›redni czas odpowiedzi serwera, zgodnoÅ›Ä‡ kodÃ³w statusu zawartych w ramach uzyskiwanych odpowiedzi, informacje o zuÅ¼yciu zasobÃ³w sprzÄ™towych serwera, czy teÅ¼ wartoÅ›Ä‡ wskaÅşnika satysfakcji klienta. Rezultaty przeprowadzonych badaÅ„ wykazaÅ‚y kluczowe znaczenie optymalizacji kodu ÅşrÃ³dÅ‚owego aplikacji, w kontekÅ›cie realizacji rozbudowanych i skalowalnych usÅ‚ug sieciowych.

Rozdział 3

Opis problemu badawczego

W ramach niniejszego rozdziaĹ‚u omÄ‚wiony zostaĹ‚, podjmowany problem badawczy. W zwiÄ…zku z jego zĹ‚oĹ›ciÄ…, autor pracy zdecydowaĹ‚, siÄ™ na podziaĹ‚, tego zagadnienia na okreĹ›lone aspekty, bÄ™dÄ…ce rÄ‚Ĺ‚norodnymi wzglÄ™dem funkcjonalnoĹ›ci internetowych interfejsÄ‚w programowania aplikacji. Na koniec tej czÄ™Ĺ›ci pracy, na podstawie sformuĹ‚owanych konktekstÄ‚w badawczych, zdefiniowano listÄ™ scenariuszy realizacji badaĹ„.

3.1. Przedstawienie aspektÄ‚w problemu badawczego

Podejmowany w ramach niniejszej pracy problem badawczy, tyczy siÄ™ wydajnoĹ›ci usĹ‚ug sieciowych, ktÄ‚rych charakterystyka wyrÄ‚Ĺ‚nia zastosowanie wielu odrÄ™bnych komponentÄ‚w programistycznych. Komponentami tymi, mogÄ… byÄ‡ zarÄ‚wno: biblioteki obsĹ‚ugujÄ…ce proces mapowania obiektowo-relacyjnego, zastosowany wzorzec projektowy w kontekĹ›cie wewnÄ™trznej architektury API, rodzaj wykorzystywanego zewnÄ™trznego Ĺşrÿdeĺ‚a danych, czy teĹ‚ wybrana chmurowa platforma wdroĹ‚eniowa. W zwiÄ…zku z mnogoĹ›ciÄ… zagadnieĹ„ wystÄ™pujÄ…cych w ramach niniejszego problemu badawczego, zdecydowano siÄ™ na dokonanie podziaĹ‚u jego opisu w taki sposÄ‚b, aby na podstawie kaĹ‚dego z aspektÄ‚w, moĹ‚liwe byĹ‚o zdefiniowanie wyspecyfikowanych scenariuszy realizacji badaĹ„.

KaĹ‚dy z aspektÄ‚w problemu badawczego wymieniony poniĹ‚ej, rozpatrywany bÄ™dzie wzglÄ™dem dwÄ‚ch odrÄ™bnych zestawÄ‚w technologii programistycznych (tj. C#/.NET oraz JavaScript/NodeJS). Scenariusze badawcze, opracowane na podstawie, kaĹ‚dego z opisanych w tym podrozdziale aspektÄ‚w, uwzglÄ™dniaÄ‡ bÄ™dÄ… porÄ‚wnanie uzyskanych wynikÄ‚w badaĹ„ dla obu wymienionych rozwiÄ…zaĹ„ informatycznych.

WydajnoĹ›Ä‡ interfejsu API wzglÄ™dem liczby Ĺ‚Ä…daĹ„ generowanych przez klientÄ‚w

Pierwszym z omawianych aspektÄ‚w rozwaĹ‚anego problemu badawczego jest wpĹ‚yw wydajnoĹ›ci dziaĹ‚ania interfejsu programowania aplikacji, wzglÄ™dem liczby klientÄ‚w, ktÄ‚rzy w sposÄ‚b rÄ‚Ĺ‚wnoległy generujÄ… Ĺ‚Ä…dania w kierunku API.

WydajnoĹ›Ä‡, w kontekĹ›cie tego wĹ‚aĹ›nie aspektu, interpretowana jest poprzez metryki czasu odpowiedzi na Ĺ‚Ä…danie, a takĹ‚e procentowe wartoĹ›ci wykorzystania zasobÄ‚w sprzÄ™towych interfejsu programowania aplikacji, takich jak centralna jednostka przetwarzania, czy teĹ‚ pamiÄ™Ä‡ operacyjna o dostÄ™pie swobodnym.

Zgodnie z obowiÄ…zujÄ…cymi praktykami realizacji pomiarÄłw wydajnoĹ›ci usĹ‚ug sieciowych, charakterystyka ta, powinna byÄ‡ wyliczana w oparciu o technikÄ™ testowania rozproszonego *(ang. Distributed Testing)*. Przedstawiana technika, zakĹ‚ada wykorzystanie wielu odrÄ™bnych systemÄłw informatycznych, wykonujÄ…cych rÄłwnolegle ewaluacjÄ™ obciÄ…Ĺ»eniowÄ…. WartoĹ›ci metryk, uzyskane dla kaĹ»dej z maszyn przeprowadzajÄ…cych testy, powinny zostaÄ‡ zgrupowane, a takĹ»e analizowane jako pochodzÄ…ce z jednego ĹşrÄłdĹ‚a.

Zmiana wydajnoĹ›ci, obserwowana powinna byÄ‡ wraz ze staĹ‚ym zwiÄ™kszeniem natÄ™Ĺ»enia liczby klientÄłw, a dla kaĹ»dego z kolejnych przedziaĹ‚Ä„w liczbowych, uwzglÄ™dniajÄ…cych kolejne przyrosty wysyĹ‚anych pakietÄłw, metryki wydajnoĹ›ciowe powinny byÄ‡ porÄłwnywane wzglÄ™dem ustalonego wskaĹşnika referencyjnego. Kalkulacja tego wskaĹşnika natomiast, wykonywana jest w oparciu o ewaluacjÄ™ wydajnoĹ›ci dla standardowych warunkÄłw pracy interfejsu programowania aplikacji. PrzykĹ‚adem takich warunkÄłw, moĹ»e byÄ‡ realizacja testu obciÄ…Ĺ»eniowego dla pojedynczej maszyny testujÄ…cej.

Jednym z podstawowych, otwartych standardÄłw, ktÄłre posĹ‚uĹ»yÄ‡ mogÄ… do budowy wskaĹşnika referencyjnego jest APDEX *(ang. Application Performance Index)*. Indeks ten, pozwala na zdefiniowanie trzech przedziaĹ‚Ä„w liczbowych, okreĹ›lających odczucia klienta testujÄ…cego oprogramowanie. PrzedziaĹ‚y te, przedstawiane sÄ… jako progi satysfakcji, tolerancji oraz frustracji. Po wykonaniu testÄłw odbywajÄ…cych siÄ™ w standardowych warunkach pracy usĹ‚ugi sieciowej, ustalenie wartoĹ›ci metryk wydajnoĹ›ciowych dla kaĹ»dego z trzech progÄłw jest moĹ»liwe, a co za tym idzie, moĹ»liwe jest takĹ»e porÄłwnanie wynikÄłw uzyskiwanych przy dowolnej liczbie klientÄłw API, wzglÄ™dem ustalonych progÄłw.

Na podstawie weryfikacji wydajnoĹ›ci interfejsu API wzglÄ™dem liczby generowanych rÄłwnolegle Ĺ»Ä…daĹ„, naleĹ»y takĹ»e ustaliÄ‡ graniczną… wartoĹ›Ä‡ sumy maszyn klienckich, dla ktÄłrych interfejs programowania aplikacji jest w stanie obsĹ‚ugiwaÄ‡ zapytania, w czasie zawierającym siÄ™ w kaĹ»dym z przedziaĹ‚Ä„w referencyjnych.

Korelacja charakterystyk wydajnoĹ›ciowych wzglÄ™dem okreĹ›lonego zewnÄ™trznego ĹşrÄłdĹ‚a danych

Kolejny z kontekstÄłw problemu badawczego dotyczy wpĹ‚ywu zastosowania odmiennych zewnÄ™trznych ĹşrÄłdeĹ‚ danych, na efektywnoĹ›Ä‡ pracy usĹ‚ugi sieciowej jakÄ… jest interfejs API.

Koncepcja wydajnoĹ›ci w przypadku tego aspektu problemu, rozumiana jest w sposÄłb analogiczny do pojÄ™cia, wprowadzonego w ramach poprzedniej sekcji.

W zwiÄ…zku z istotnoĹ›ciÄ… uwzgladnienia zewnÄ™trznych ĹşrÄłdeĹ‚ danych, jako elementÄłw z ktÄłrymi nieustannie komunikujÄ… siÄ™ nowoczesne usĹ‚ugi sieciowe, w ramach niniejszego aspektu omawianego problemu badawczego, weryfikowany jest wpĹ‚yw sposobu obsĹ‚ugi najpopularniejszych spoĹ›rÄłd dostÄ™pnych nieodpĹ‚atnie systemÄłw bazodanowych, na efektywnoĹ›Ä‡ operacji realizowanych przez poszczegÄłlne interfejsy programowania aplikacji. Problem badawczy, uwzglÄ™dnia zastosowanie zarÄłwno czterech relacyjnych systemÄłw baz danych, jak i jednego nierelacyjnego.

SposÄłb obserwacji zmiany wydajnoĹ›ci usĹ‚ugi sieciowej, rÄłwnieĹ» jest analogiczny, do tego, ktÄłry zostaĹ‚, przedstawiony w ramach poprzedniego aspektu, jednakĹ»e naleĹ»y zwrÄłciÄ‡ uwagÄ™, na koniecznoĹ›Ä‡ ponownego zdefiniowania przedziaĹ‚Ä„w satysfakcji, tolerancji oraz frustracji dla wskaĹşnika referencyjnego. Referencja do wartoĹ›ci tego wskaĹşnika, obliczonego bez uwzglÄ™dnienia poĹ‚Ä…czenia z systemem bazodanowym, prowadziĹ‚a by do zaniĹ»enia wartoĹ›ci ogÄłlnej wydajnoĹ›ci testowanego oprogramowania.

Kluczowym czynnikiem omawianego aspektu problemu badawczego jest zapewnienie deterministycznego charakteru stanu Ĺ‚Ă…cza sieciowego, wystÄ™pujÄ…cego pomiÄ™dzy serwerem bazodanowym a usĹ‚ugÄ… interfejsu programowania aplikacji. Dlatego teĹ‚, w przypadku testĂłw realizowanych w Ĺ›rodowisku lokalnym, oba komponenty programowo-sprzÄ™towe powinny znajdowaÄ‡ siÄ™ w tej samej lokalizacji. NaleĹ‚y podkreĹ›liÄ‡ jednak, Ĺ‚e waĹ‚nym jest, aby obie usĹ‚ugi informatyczne nie byĹ‚y uruchomione w ramach tego samego Ĺ›rodowiska sprzÄ™towego. DziÄ™ki temu, pozyskane wartoĹ›ci metryk wykorzystania zasobĂłw sprzÄ™towych nie bÄ™dÄ… obarczone niedokĹ‚adnoĹ›ciÄ…. W kontekĹ›cie realizacji badaĹ„ w Ĺ›rodowisku chmurowym natomiast, waĹ‚nym jest wdroĹ‚enie zarĂłwno API, jak i serwera bazy danych, w ramach tego samego centrum obliczeniowego, a takĹ‚e rozdzielenie obu usĹ‚ug, pomiÄ™dzy odmienne fizyczne urzÄ…dzenia. Ponadto, modyfikacji powinien ulec jeden z elementĂłw kryterium wydajnoĹ›ci, ktĂłry definiowany jest jako czas odpowiedzi interfejsu na Ĺ‚Ä…danie klienta. W zwiÄ…zku z dyspersjÄ… geograficznÄ… obu stron komunikacji, niemoĹ‚liwym jest zachowanie przewidywalnego charakteru Ĺ‚Ă…cza sieciowego wykorzystywanego do transmisji danych. Twierdzenie to, implikuje koniecznoĹ›Ä‡ realizacji pomiaru czasu dziaĹ‚ania usĹ‚ugi sieciowej w sposĂłb odmienny. Kryterium czasu odpowiedzi na Ĺ‚Ä…danie, rozumiane w tym przypadku jest jako przedziaĹ‚ czasowy od momentu pozyskania Ĺ‚Ä…dania, do momentu zakoĹ„czenia wszystkich operacji, realizowanych w kontekĹ›cie tego Ĺ‚Ä…dania.

EfektywnoĹ›Ä‡ realizacji zĹ‚oĹ‚onych obliczeĹ„ oraz wsparcia dla programowania wspĂłĹ‚bieĹ‚nego i metod asynchronicznych

Niniejszy aspekt problemu badawczego dotyczy wpĹ‚ywu wykorzystania, dostÄ™pnych w ramach okreĹ›lonego jÄ™zyka mechanizmĂłw programowania wspĂłĹ‚bieĹ‚nego, a takĹ‚e sposobu realizacji operacji asynchronicznych, na efektywnoĹ›Ä‡ przeprowadzania kalkulacji w obrÄ™bie warstwy logiki biznesowej interfejsu programowania aplikacji.

PojÄ™cie efektywnoĹ›ci dokonywanych kalkulacji postrzegane jest poprzez liczbÄ™ wykonanych iteracji gĹ‚Ăłwnej pÄ™tli zaimplementowanego algorytmu metaheurystycznego, rozwiÄ…zujÄ…cego okreĹ›lony problem z rodziny NP-trudnych.

W celu zachowania rzetelnoĹ›ci badaĹ„, omawiany fragment problemu badawczego uwzglÄ™dnia zastosowanie analogicznego algorytmu, realizujÄ…cego operacje w ten sam sposĂłb, a takĹ‚e rozwiÄ…zujÄ…cego ten sam problem obliczeniowy. W tym przypadku, zaobserwowaÄ‡ bÄ™dzie moĹ‚na fakt przystosowania technologii poddawanej ewaluacji, do dokonywania procesu zrĂłwnoleglania obliczeĹ„, a takĹ‚e przeprowadzania wewnÄ™trznej optymalizacji okreĹ›lonych linii zdefiniowanego kodu Ĺşrâ€¦dĹ‚owego.

Zaimplementowany algorytm metaheurystyczny, dostÄ™pny bÄ™dzie bezpoĹ›rednio z poziomu punktĂłw koĹ„cowych badanych interfejsĂłw programowania aplikacji, a liczba iteracji gĹ‚Ăłwnej pÄ™tli algorytmu, mierzona bÄ™dzie dla ustalonego, staĹ‚ego czasu wykonania programu.

Ponadto, omawiany aspekt badawczy dotyczy takĹ‚e weryfikacji wydajnoĹ›ci w kontekĹ›cie zastosowania metod asynchronicznych. W zwiÄ…zku ze znaczÄ…co odmiennÄ… strukturÄ… badanych Ĺ›rodowisk uruchomieniowych oraz jÄ™zykĂłw programowania, mechanizmy obsĹ‚ugi operacji asynchronicznych zaimplementowane sÄ… w tych technologiach, na rĂłĹ‚Ĺ‚nych poziomach obsĹ‚ugi programu. W jedym przypadku, obsĹ‚uga operacji tych, jest wykonywana bezpoĹ›rednio w ramach jÄ™zyka programowania, natomiast w kontekĹ›cie drugiej z technologii, metody ktĂłrych wynik nie jest dostarczany natychmiastowo, muszÄ… zostaÄ‡ obsĹ‚uĹ‚one wewnÄ…trz Ĺ›rodowiska uruchomieniowego.

Badanie weryfikacji wydajnoĹ›ci dla funkcji asynchronicznych, oparte jest o odwoĹ‚anie siÄ™ interfejsu API, do wspĂłĹ‚pracujÄ…cej z nim hipertekstowej usĹ‚ugi sieciowej,

peĹ‚niÄ…cej rolÄ™ poĹ›rednika w dostÄ™pie do zdefiniowanych wewnÄ…trz niej informacji. Przedstawiona usĹ‚uga sieciowa, zostanie zaimplementowana jako odrÄ™bne oprogramowanie i bÄ™dzie niezaleĹĽna od obu porĂłwnywanych technologii.

W kontekĹ›cie drugiej z czÄ™Ĺ›ci aspektu problemu badawczego, metrykÄ… wydajnoĹ›ci bÄ™dzie czas odpowiedzi interfejsu na ĹĽÄ…danie.

WpĹ‚yw zastosowania wzorca projektowego podziaĹ‚u odpowiedzialnoĹ›ci na efektywnoĹ›Ä‡ realizacji operacji bazodanowych

RozwaĹĽany aspekt problemu badawczego tyczy siÄ™ wpĹ‚ywu implementacji optymalizacji wydajnoĹ›ciowych w kontekĹ›cie komunikacji interfejsu programowania aplikacji z zewnÄ™trznym ĹşrĂłdĹ‚em danych.

WykorzystujÄ…c konwencjonalnÄ… trĂłjwarstwowÄ… architekturÄ™ interfejsu API, stosowany zostaje ten sam model danych, zarĂłwno do operacji odczytu jak i zapisu. Powoduje to brak moĹĽliwoĹ›ci dostosowania modelu, wzglÄ™dem specyfiki konkretnego rodzaju operacji. Wprowadzenie wzorca projektowego separacji zapytaĹ„ oraz komend ma na celu umoĹĽliwienie dokonania optymalizacji wydajnoĹ›ciowych wyizolowanych fragmentĂłw modelu danych, a takĹĽe wykorzystywanie ich tylko i wyĹ‚Ä…cznie w kontekĹ›cie jednego typu operacji.

Co wiÄ™cej, optymalizacja moĹĽe byÄ‡ wykonana nie tylko na poziomie modelu danych, ale takĹĽe w ramach fizyczynych struktur zawartych wewnÄ…trz obsĹ‚ugiwanego systemu bazodanowego. Dlatego teĹĽ, przedstawiany aspekt problemu badawczego dotyczy zastosowania zarĂłwno odrÄ™bnych modeli danych wewnÄ…trz API, odrÄ™bnych struktur programistycznych obsĹ‚ugujÄ…cych dane modelu, jak i odseparowanych zewnÄ™trznych ĹşrĂłdeĹ‚, danych.

Oba zastosowane ĹşrĂłdĹ‚a danych, powinny cechowaÄ‡ siÄ™ takÄ… samÄ… strukturÄ…, jednakĹĽe kaĹĽde z nich powinno wprowadzaÄ‡ charakterystyczne dla typu wykonywanych operacji, usprawnienia wydajnoĹ›ciowe. Ponadto, aby zachowaÄ‡ spĂłjnoĹ›Ä‡ zawartoĹ›ci dostÄ™pnej dla klienta w ramach API, po odwoĹ‚aniu siÄ™ do systemu bazodanowego w celu zapisania rekordu, musi on zostaÄ‡ nastÄ™pnie zreplikowany do ĹşrĂłdĹ‚a danych obsĹ‚ugujÄ…cego operacjÄ™ odczytu.

WydajnoĹ›Ä‡, rozumiana poprzez czas odpowiedzi interfejsu API na ĹĽÄ…danie klienta, powinna zostaÄ‡ porĂłwnana z tÄ…, wykazywanÄ… przez usĹ‚ugÄ™ sieciowÄ… opierajÄ…cÄ… siÄ™ na architekturze 3-warstwowej i wykorzystujÄ…cÄ… pojedyncze ĹşrĂłdĹ‚o danych.

WpĹ‚yw zastosowania mechanizmĂłw pamiÄ™ci podrÄ™cznej na wydajnoĹ›Ä‡ interfejsĂłw API

Problem badawczy w kontekĹ›cie wykorzystania mechanizmĂłw pamiÄ™ci podrÄ™cznej, dotyczy porĂłwnania efektywnoĹ›ci dziaĹ‚ania interfejsĂłw programowania aplikacji implementujÄ…cych standardowy oraz autorski mechanizm przechowywania rezultatĂłw wykonanych uprzednio ĹĽÄ…daĹ„.

Standardowy mechanizm przechowywania ĹĽÄ…daĹ„ w ramach pamiÄ™ci podrÄ™cznej uwzglÄ™dniaÄ‡ powinien staĹ‚y czas waĹĽnoĹ›ci pojedynczego wpisu, a takĹĽe jego uniewaĹĽnienie w przypadku wykonania operacji modyfikujÄ…cej dane. W takim przypadku, czas odpowiedzi na ĹĽÄ…danie powinien byÄ‡ zwiÄ™kszony w momencie koniecznoĹ›ci odwoĹ‚ania siÄ™ API do zewnÄ™trznego ĹşrĂłdĹ‚a danych, a nastÄ™pnie zredukowany w przedziale czasowym, w ramach ktĂłrego wpis pamiÄ™ci podrÄ™cznej jest aktywny.

Zaimplementowany autorski mechanizm pamiÄ™ci podrÄ™cznej wyrĂłĹĽniaÄ‡ bÄ™dzie siÄ™ zmiennym czasem waĹĽnoĹ›ci poszczegĂłlnych wpisĂłw, ktĂłry zaleĹĽny bÄ™dzie od prawdopodobieĹ„stwa wywoĹ,ania okreĹ›lonego punktu koĹ„cowego, na podstawie informacji o liczbie historycznych wywoĹ,aĹ„. Czym wiÄ™ksze istnieje prawdopodobieĹ„stwo ponownego wywoĹ,ania punktu koĹ„cowego, tym czas waĹĽnoĹ›ci rezultatu przechowywanego w pamiÄ™ci podrÄ™cznej bÄ™dzie wiÄ™kszy. Analogicznie do standardowego mechanizmu pamiÄ™ci podrÄ™cznej, operacja modyfikacji danych uniewaĹĽnia wszystkie spoĹ›rĂłd wpisĂłw, ktĂłre odwoĹ,ujÄ… siÄ™ do przeksztaĹ,conych informacji.

Celem niniejszego aspektu badawczego w ramach rozwaĹĽanego problemu jest porĂłwnanie zmiany wydajnoĹ›ci dziaĹ,ania API, postrzeganej jako Ĺ›redni czas odpowiedzi na ĹĽÄ…danie w ustalonym, staĹ,ym przedziale czasowym. PorĂłwnywane zostanÄ… mechanizmy standardowy oraz autorski, odrÄ™bnie dla kaĹĽdej z technologii programistycznych.

WpĹ,yw wdroĹĽenia interfejsu API na dedykowanej platformie chmurowej na jego efektywnoĹ›Ä‡ dziaĹ,ania

Ostatni z przedstawianych aspektĂłw problemu badawczego odnosi siÄ™ do wpĹ,ywu wydajnoĹ›ci pracy interfejsu API, w zaleĹĽnoĹ›ci od rodzaju zatosowanej platformy chmurowej, na jakiej zostanie on wdroĹĽony.

Interfejs programowania aplikacji jest usĹ,ugÄ…, ktĂłrej funkcjonowanie jest nieodĹ,Ä…cznie powiÄ…zane z serwerem internetowym. Serwer sieci Web peĹ,ni rolÄ™ warstwy opakowujÄ…cej, wewnÄ…trz ktĂłrej dziaĹ,aÄ‡ moĹĽe interfejs API. WdroĹĽenie rozwaĹĽanej usĹ,ugi sieciowej w ramach sieci Internet, wiÄ…ĹĽe siÄ™ w zwiÄ…zku z tym z uruchomieniem serwera sieci web w ramach komputera eksponowanego w sieci rozlegĹ,ej. NaleĹĽy rĂłwnieĹĽ zauwaĹĽyÄ‡ koniecznoĹ›Ä‡ zastosowania hipertekstowego serwera poĹ›redniczÄ…cego, po to, aby klient usĹ,ugi, mĂłgĹ, siÄ™ z niÄ… komunikowaÄ‡ z wykorzystaniem protokoĹ,u HTTP.

System informatyczny, skĹ,adajÄ…cy siÄ™ z przedstawionych powyĹĽej komponentĂłw moĹĽe zostaÄ‡ uruchomiony w ramach wirtualnego serwera prywatnego, udostÄ™pnianego przez okreĹ›lonego dostawcÄ™ infrastruktury serwerowej. Model taki, definiowany jest jako infrastruktura w postaci usĹ,ugi klienckiej *(ang. Infrastructure as a Service)*. Ponadto, przygotowane oprogramowanie moĹĽe zostaÄ‡ wdroĹĽone na dedykowanej okreĹ›lonej technologii, platformie chmurowej. W ramach platformy tej, uĹĽytkownik, za pomocÄ… dostarczonego interfejsu komunikacji moĹĽe wdraĹĽaÄ‡ oraz konfigurowaÄ‡ dziaĹ,anie swojego oprogramowania. Taki model dostarczania zasobu z kolei, nazywany jest platformÄ… w postaci usĹ,ugi klienckiej *(ang. Platform as a Service)*.

Niniejszy aspekt problemu badawczego, dotyczy porĂłwnania wydajnoĹ›ci API, w zaleĹĽnoĹ›ci od jego wdroĹĽenia na generycznym wirtualnym serwerze prywatnym, a takĹĽe dedykowanej platformie chmurowej, dostosowanej pod kÄ…tem okreĹ›lonego Ĺ›rodowiska uruchomieniowego oraz jÄ™zyka programowania.

WskaĹşnik ewaluacji efektywnoĹ›ci dziaĹ,ania interfejsu programowania aplikacji, obejmuje te same metryki, ktĂłre przedstawione zostaĹ,y w pierwszym spoĹ›rĂłd omawianych aspektĂłw problemu badawczego. Kryterium czasu odpowiedzi na ĹĽÄ…danie, musi zostaÄ‡ jednakĹĽe uniezaleĹĽnione od niedeterministycznego charakteru Ĺ,Ä…cza internetowego, dlatego teĹĽ, ten wĹ,aĹ›nie parametr, bÄ™dzie dotyczyĹ, czasu od momentu otrzymania ĹĽÄ…dania przez API, do chwili wygenerowania odpowiedzi na ĹĽÄ…danie.

3.2. SformuĹ‚owanie scenariuszy badawczych

Na podstawie przedstawionych w poprzednim podrozdziale aspektĂłw problemu badawczego, sformuĹ‚owane zostaĹ‚y konkretne scenariusze badawcze. W kaĹĽdym ze scenariuszy, zdefiniowano zbiĂłr czynnoĹ›ci wykonywanych w ramach okreĹ›lonego badania, wyszczegĂłlniono kryteria porĂłwnawcze dla danej obserwacji, wymieniono dostosowywalne parametry badania, a takĹĽe skonkretyzowano czynnoĹ›ci, ktĂłre muszÄ… zostaÄ‡ podjÄ™te, jako warunki konieczne przed wykonaniem badania. KaĹĽdy ze scenariuszy badawczych odzwierciedlony zostaĹ‚, w formie tabeli.

[c]|llll| Scenariusz badawczy - badanie przeprowadzone w kontekĹ›cie systemĂłw bazodanowych

Nazwa scenariusza badawczego:	
WpĹ‚yw zastosowanego systemu bazodanowego na efektywnoĹ›Ä‡ dziaĹ‚ania interfejsu programowania aplikacji przy zmiennej liczbie ĹĽÄ…daĹ„ klientĂłw	
Topologia fizyczna Ĺ›rodowiska badawczego:	
Konfiguracja pierwsza lokalnego Ĺ›rodowiska badawczego **??**	
CzynnoĹ›ci implementacyjne:	
• Zaimplementowanie interfejsĂłw programowania aplikacji w technologiach C#/.NET oraz NodeJS/ExpressJS. • Konfiguracja interfejsĂłw programowania aplikacji w celu obsĹ‚ugi systemĂłw bazodanowych: Microsoft SQL Server, MySQL, PostgreSQL, SQLite oraz MongoDB. • Konfiguracja topologii fizycznej Ĺ›rodowiska badawczego. • Konfiguracja narzÄ™dzia do wykonywania testĂłw wydajnoĹ›ciowych.	
CzynnoĹ›ci badawcze:	
• Realizacja testĂłw wydajnoĹ›ciowych z uwzglÄ™dnieniem zmiennoĹ›ci liczby klientĂłw. • Obserwacja oraz gromadzenie wartoĹ›ci pomiarowych dotyczÄ…cych kryteriĂłw porĂłwnawczych. • Dostosowywanie wartoĹ›ci parametrĂłw przeprowadzanego badania. • Analiza statystyczna otrzymanych wynikĂłw.	
Warunki poczÄ…tkowe podjÄ™cia czynnoĹ›ci badawczych:	
Przed realizacjÄ… testĂłw wydajnoĹ›ciowych zagwarantowana zostanie poprawnoĹ›Ä‡ dziaĹ‚ania kaĹĽdego z interfejsĂłw programowania aplikacji poprzez wykonanie ewaluacji funkcjonalnej.	

Opis scenariusza badawczego:

Po wykonaniu konfiguracji topologii fizycznej Ĺ›rodowiska badawczego, urzÄ…dzenia klienckie bÄ™dÄ… wysyĹ‚aÄ‡ Ĺ»Ä…dania protokoĹ‚u hipertekstowego w kierunku interfejsu programowania aplikacji. Liczba Ĺ»Ä…daĹ„ tych bÄ™dzie sukcesywnie zwiÄ™kszana, poprzez uruchamianie kolejnych rÄ…wnolegle pracujÄ…cych wÄ…tkÄ…w oprogramowania testowego. Dla kaĹ»dej wartoĹ›ci liczby urzÄ…dzeĹ„ klienckich, badanie zostanie wykonane dziesiÄ™ciokrotnie, a uzyskane wyniki zostanÄ… nastÄ™pnie uĹ›rednione. Odpowiednie przedziaĹ‚y wartoĹ›ci omawianego parametru wyznaczaÄ‡ bÄ™dÄ… granicÄ™ pomiÄ™dzy testami linii bazowej, obciÄ…Ĺ»eniowymi, a takĹ»e przeciÄ…Ĺ»ajÄ…cymi. Badanie zostanie powtÄ…rzone dla kaĹ»dego z rozwaĹ»anych systemÄ…w bazodanowych w obrÄ™bie obu porÄ…wnywanych technologii programistycznych. Po zgromadzeniu wynikÄ…w badaĹ„ przeprowadzone zostanÄ… parowe testy statystyczne wykazujÄ…ce istotnoĹ›Ä‡ rÄ…Ĺ»nic pomiarowych.

Kryteria porÄ…wnawcze:

- Czas odpowiedzi interfejsu programowania aplikacji na Ĺ»Ä…danie klienta.
- Procent wykorzystania centralnej jednostki przetwarzania serwera, na ktÄ…rym uruchomiony zostaĹ‚, interfejs programowania aplikacji.
- IloĹ›Ä‡ wykorzystanej pamiÄ™ci operacyjnej serwera, na ktÄ…rym uruchomiony zostaĹ‚, interfejs programowania aplikacji.

Parametry badania:

- Liczba klientÄ…w rÄ…wnolegle wysyĹ‚ających Ĺ»Ä…dania.
- Technologia programistyczna zastosowana do implementacji interfejsu programowania aplikacji.
- Rodzaj systemu bazodanowego komunikującego siÄ™ z interfejsem programowania aplikacji.

[c]|llll| Scenariusz badawczy - badanie przeprowadzone w kontekĹ›cie realizacji operacji wspÄ…ĹˇĹ‚ bieĹ»Ä…cych

Nazwa scenariusza badawczego:

WpĹ‚yw zastosowanej technologii programistycznej na wydajnoĹ›Ä‡ realizacji operacji wspÄ…ĹˇĹ‚ bieĹ»Ä…cych

Topologia fizyczna Ĺ›rodowiska badawczego:

Konfiguracja druga lokalnego Ĺ›rodowiska badawczego **??**

CzynnoĹ›ci implementacyjne:

- Zaimplementowanie genetycznego algorytmu metaheurystycznego dla symetrycznego problemu komiwojaĹĽera w jÄ™zykach programowania C# oraz JavaScript.
- Zaimplementowanie mechanizmĂłw pomiaru czasu wykonania algorytmu
- Konfiguracja interfejsĂłw programowania aplikacji w celu obsĹ‚ugi algorytmĂłw metaheurystycznych z poziomu punktu koĹ„cowego API.
- Konfiguracja topologii fizycznej Ĺ›rodowiska badawczego.
- Konfiguracja narzÄ™dzia do wykonywania testĂłw wydajnoĹ›ciowych.

CzynnoĹ›ci badawcze:

- Realizacja testĂłw wydajnoĹ›ciowych dla porĂłwnywanych technologii programistycznych.
- Obserwacja oraz gromadzenie wartoĹ›ci pomiarowych dotyczÄ…cych kryteriĂłw porĂłwnawczych.
- Dostosowywanie wartoĹ›ci parametrĂłw przeprowadzanego badania.
- Analiza statystyczna otrzymanych wynikĂłw.

Warunki poczÄ…tkowe podjÄ™cia czynnoĹ›ci badawczych:

Przed realizacjÄ… testĂłw wydajnoĹ›ciowych zagwarantowana zostanie poprawnoĹ›Ä‡ dziaĹ‚ania kaĹĽdego z algorytmĂłw metaheurystycznych. Ponadto, kod ĹşrĂłdĹ‚owy programĂłw implementujÄ…cych algorytmy zostanie przeksztaĹ‚cony w taki sposĂłb, aby niezaleĹĽnie od jÄ™zyka programowania, realizowaĹ‚ operacje w sposĂłb analogiczny.

Opis scenariusza badawczego:

Po wykonaniu konfiguracji topologii fizycznej Ĺ›rodowiska badawczego, pojedyncze urzÄ…dzenie klienckie bÄ™dzie wysyĹ‚aĂ‡ z ustalonÄ… czÄ™stotliwoĹ›ciÄ…, ĹĽÄ…dania wykonania algorytmu. Po odebraniu zapytania od klienta, algorytm bÄ™dzie uruchamiany, a czas trwania obliczeĹ„ bÄ™dzie zawsze wartoĹ›ciÄ… staĹ‚Ä…. W trakcie wykonywanych kalkulacji zliczana bÄ™dzie liczba iteracji gĹ‚Ăłwnej pÄ™tli kodu algorytmu. Liczba ta, stanowiÄ‡ bÄ™dzie kryterium porĂłwnawcze. Dla kaĹĽdej z porĂłwnywanych technologii programistycznych wykonana zostanie seria dwudziestu cyklicznych ĹĽÄ…daĹ„ klienta. Po zgromadzeniu wynikĂłw badaĹ„ przeprowadzone zostanÄ… parowe testy statystyczne wykazujÄ…ce istotnoĹ›Ä‡ rĂłĹĽnic pomiarowych.

Kryteria porĂłwnawcze:

- Liczba iteracji gĹ‚Ăłwnej pÄ™tli algorytmu metaheurystycznego.

Parametry badania:

- CzÄ™stotliwoĹ›Ä‡ generowania ĹĽÄ…daĹ„ protokoĹ‚u hipertekstowego.
- Technologia programistyczna zastosowana do implementacji interfejsu programowania aplikacji.

[c]|llll| Scenariusz badawczy - badanie przeprowadzone w kontekĹ›cie obsĹ‚ugi operacji asynchronicznych

Nazwa scenariusza badawczego:	

WpĹ‚yw zastosowanej technologii programistycznej na efektywnoĹ›Ä‡ obsĹ‚ugi operacji asynchronicznych

Topologia fizyczna Ĺ›rodowiska badawczego:	

Konfiguracja trzecia lokalnego Ĺ›rodowiska badawczego **??**

CzynnoĹ›ci implementacyjne:	

- Zaimplementowanie interfejsÓłw programowania aplikacji w technologiach C#/.NET, NodeJS/ExpressJS oraz Python/Flask.
- Zdefiniowanie punktÓłw koĹ„cowych odpowiedzialnych za komunikacjÄ™ badanego interfejsu z zewnÄ™trznym API.
- Konfiguracja topologii fizycznej Ĺ›rodowiska badawczego.
- Konfiguracja narzÄ™dzia do wykonywania testÓłw wydajnoĹ›ciowych.

CzynnoĹ›ci badawcze:	

- Realizacja testÓłw wydajnoĹ›ciowych dla porÓłwnywanych technologii programistycznych.
- Obserwacja oraz gromadzenie wartoĹ›ci pomiarowych dotyczÄ…cych kryteriÓłw porÓłwnawczych.
- Dostosowywanie wartoĹ›ci parametrÓłw przeprowadzanego badania.
- Analiza statystyczna otrzymanych wynikÓłw.

Warunki poczÄ…tkowe podjÄ™cia czynnoĹ›ci badawczych:	

Przed realizacjÄ… testÓłw wydajnoĹ›ciowych zagwarantowane zostanie poprawne poĹ‚Ä…czenie pomiÄ™dzy kaĹĽdym z badanych interfejsÓłw programowania aplikacji a zewnÄ™trznÄ… usĹ‚ugÄ… sieciowÄ…. Co wiÄ™cej, zweryfikowana zostanie poprawnoĹ›Ä‡ implementacji funkcjonalnoĹ›ci zewnÄ™trznej usĹ‚ugi sieciowej (tj. interfejsu API zaimplementowanego w jÄ™zyku Python)

Opis scenariusza badawczego:	

Po wykonaniu konfiguracji topologii fizycznej Ĺ›rodowiska badawczego, urzÄ…dzenia klienckie bÄ™dÄ… rÄ…wnolegle wysyĹ‚aÄ‡ ĹĽÄ…dania http w kierunku interfejsu programowania aplikacji. Po odebraniu zapytania od klienta, interfejs API, Ĺ‚Ä…czyÄ‡ siÄ™ bÄ™dzie ze znajdujÄ…cÄ… siÄ™ w obrÄ™bie sieci lokalnej zewnÄ™trznÄ… usĹ‚ugÄ… sieciowÄ…. Taki rodzaj operacji, kiedy strona wywoĹ‚ujÄ…ca zleca wykonanie zadania, a odpowiedĹş na to zlecenie przyjĹ›Ä‡ moĹĽe w dowolnym momencie, nazywamy operacjÄ… asynchronicznÄ…. W ramach punktu koĹ„cowego ewaluowanego API, wykonane zostanÄ… cztery operacje asynchroniczne dotyczÄ…ce rÄŁĹĽnych sposobÄŁw operowania na danych. Po uzyskaniu odpowiedzi na wszystkie z czterech operacji, interfejs programowania aplikacji bÄ™dzie zwracaĹ‚ odpowiedĹş zawierajÄ…cÄ… informacjÄ™ o stopniu poprawnoĹ›ci wykonania zleconych operacji, a takĹĽe czas odpowiedzi na ĹĽÄ…danie. Tak zdefiniowane badanie, bÄ™dzie wykonywane dla okreĹ›lonej liczby klientÄŁw rÄŁwnolegle generujÄ…cych ĹĽÄ…dania. Po zgromadzeniu wynikÄŁw badaĹ„ przeprowadzone zostanÄ… parowe testy statystyczne wykazujÄ…ce istotnoĹ›Ä‡ rÄŁĹĽnic pomiarowych.

Kryteria porÄŁwnawcze:	

- Czas odpowiedzi interfejsu programowania aplikacji na ĹĽÄ…danie klienta.
- Procent poprawnoĹ›ci wykonania zleconych operacji asynchronicznych.

Parametry badania:	

- Liczba klientÄŁw rÄŁwnolegle wysyĹ‚ajÄ…cych ĹĽÄ…dania.
- Technologia programistyczna zastosowana do implementacji interfejsu programowania aplikacji.

[c]|llll| Scenariusz badawczy - badanie przeprowadzone w kontekĹ›cie zastosowania wzorca projektowego CQRS

Nazwa scenariusza badawczego:	

WpĹ‚yw implementacji wzorca projektowego podziaĹ‚u odpowiedzialnoĹ›ci na wydajnoĹ›Ä‡ obsĹ‚ugi ĹĽÄ…dania klienta

Topologia fizyczna Ĺ›rodowiska badawczego:	

Konfiguracja pierwsza lokalnego Ĺ›rodowiska badawczego **??**

CzynnoĹ›ci implementacyjne:	

- Zaimplementowanie interfejsĂłw programowania aplikacji wykorzystujÄ…cych wzorzec projektowy CQRS w technologiach C#/.NET oraz NodeJS/ExpressJS.
- Konfiguracja interfejsĂłw programowania aplikacji w celu obsĹ‚ugi systemĂłw bazodanowych: Microsoft SQL Server, MySQL, PostgreSQL, SQLite oraz MongoDB.
- Wprowadzenie usprawnieĹ„ wydajnoĹ›ciowych dotyczÄ…cych odczytu i zapisu dla kaĹĽdego z systemĂłw bazodanowych
- PoĹ‚Ä…czenie interfejsu API z dwoma systemami bazodanowymi tego samego typu i konfiguracja automatycznej replikacji danych w momencie ich zapisu
- Konfiguracja topologii fizycznej Ĺ›rodowiska badawczego.
- Konfiguracja narzÄ™dzia do wykonywania testĂłw wydajnoĹ›ciowych.

CzynnoĹ›ci badawcze:

- Realizacja testĂłw wydajnoĹ›ciowych z uwzglÄ™dnieniem zmiennoĹ›ci liczby klientĂłw.
- Obserwacja oraz gromadzenie wartoĹ›ci pomiarowych dotyczÄ…cych kryteriĂłw porĂłwnawczych.
- Dostosowywanie wartoĹ›ci parametrĂłw przeprowadzanego badania.
- Analiza statystyczna otrzymanych wynikĂłw.

Warunki poczÄ…tkowe podjÄ™cia czynnoĹ›ci badawczych:

Przed realizacjÄ… testĂłw wydajnoĹ›ciowych zagwarantowana zostanie poprawnoĹ›Ä‡ dziaĹ‚ania kaĹĽdego z interfejsĂłw programowania aplikacji poprzez wykonanie ewaluacji funkcjonalnej.

Opis scenariusza badawczego:

Po wykonaniu konfiguracji topologii fizycznej Ĺ›rodowiska badawczego, urzÄ…dzenia klienckie bÄ™dÄ… wysyĹ‚aÄ‡ ĹĽÄ…dania protokoĹ‚u hipertekstowego w kierunku interfejsu programowania aplikacji. Liczba ĹĽÄ…daĹ„ tych bÄ™dzie sukcesywnie zwiÄ™kszana, poprzez uruchamianie kolejnych rĂłwnolegle pracujÄ…cych wÄ…tkĂłw oprogramowania testowego. Dla kaĹĽdej wartoĹ›ci liczby urzÄ…dzeĹ„ klienckich, badanie zostanie wykonane dziesiÄ™ciokrotnie, a uzyskane wyniki zostanÄ… nastÄ™pnie uĹ›rednione. Odpowiednie przedziaĹ‚y wartoĹ›ci omawianego parametru wyznaczaÄ‡ bÄ™dÄ… granicÄ™ pomiÄ™dzy testami linii bazowej, obciÄ…ĹĽeniowymi, a takĹĽe przeciÄ…ĹĽajÄ…cymi. Badanie zostanie powtĂłrzone dla kaĹĽdego z rozwaĹĽanych systemĂłw bazodanowych w obrÄ™bie obu porĂłwnywanych technologii programistycznych. Po zgromadzeniu wynikĂłw badaĹ„ przeprowadzone zostanÄ… parowe testy statystyczne wykazujÄ…ce istotnoĹ›Ä‡ rĂłĹĽnic pomiarowych. Obserwacje uzyskane w ramach niniejszego badania, porĂłwnane zostanÄ… z tymi, ustalonymi na podstawie pierwszego scenariusza badawczego **??**.

Kryteria porĂłwnawcze:

- Czas odpowiedzi interfejsu programowania aplikacji na ĹĽÄ…danie klienta.
- Procent wykorzystania centralnej jednostki przetwarzania serwera, na ktĂłrym uruchomiony zostaĹ‚, interfejs programowania aplikacji.
- IloĹ›Ä‡ wykorzystanej pamiÄ™ci operacyjnej serwera, na ktĂłrym uruchomiony zostaĹ‚, interfejs programowania aplikacji.

Parametry badania:	

- Liczba klientĂłw rĂłwnolegle wysyĹ‚ajÄ…cych ĹĽÄ…dania.
- Technologia programistyczna zastosowana do implementacji interfejsu programowania aplikacji.
- Rodzaj systemu bazodanowego komunikujÄ…cego siÄ™ z interfejsem programowania aplikacji.

[c]|llll| Scenariusz badawczy - badanie przeprowadzone w kontekĹ›cie wykorzystania mechanizmĂłw pamiÄ™ci podrÄ™cznej

Nazwa scenariusza badawczego:	

PorĂłwnanie efektywnoĹ›ci obsĹ‚ugi ĹĽÄ…daĹ„ klienckich w staĹ‚ym czasie, uwzglÄ™dniajÄ…c odmienne implementacje mechanizmĂłw pamiÄ™ci podrÄ™cznej

Topologia fizyczna Ĺ›rodowiska badawczego:	

Konfiguracja druga lokalnego Ĺ›rodowiska badawczego **??**

CzynnoĹ›ci implementacyjne:	

- Zaimplementowanie interfejsĂłw programowania aplikacji w technologiach C#/.NET oraz NodeJS/ExpressJS.
- Zaimplementowanie mechanizmĂłw pamiÄ™ci podrÄ™cznej (rozwiÄ…zanie autorskie oraz standardowe) w oparciu o bibliotekÄ™ Redis.
- Konfiguracja interfejsĂłw programowania aplikacji w celu obsĹ‚ugi systemu bazodanowego MySQL.
- Konfiguracja topologii fizycznej Ĺ›rodowiska badawczego.
- Konfiguracja narzÄ™dzia do wykonywania testĂłw wydajnoĹ›ciowych.

CzynnoĹ›ci badawcze:	

- Realizacja testĂłw wydajnoĹ›ciowych z uwzglÄ™dnieniem zmiennoĹ›ci czasu trwania testu.
- Obserwacja oraz gromadzenie wartoĹ›ci pomiarowych dotyczÄ…cych kryteriĂłw porĂłwnawczych.
- Dostosowywanie wartoĹ›ci parametrĂłw przeprowadzanego badania.
- Analiza statystyczna otrzymanych wynikĂłw.

Warunki poczÄ…tkowe podjÄ™cia czynnoĹ›ci badawczych:

Przed realizacjÄ… testĂłw wydajnoĹ›ciowych zagwarantowana zostanie poprawnoĹ›Ä‡ dziaĹ‚ania kaĹĽdego z interfejsĂłw programowania aplikacji, a takĹĽe kaĹĽdego z zaimplementowanych mechanizmĂłw pamiÄ™ci podrÄ™cznej. Do autorskiego mechanizmu pamiÄ™ci cache dostarczone zostanÄ… przygotowane informacje historyczne, ktĂłrych zawartoĹ›Ä‡ dotyczy czÄ™stotliwoĹ›ci wywoĹ‚ywania okreĹ›lonych punktĂłw koĹ„cowych interfejsu programowania aplikacji.

Opis scenariusza badawczego:

Po wykonaniu konfiguracji topologii fizycznej Ĺ›rodowiska badawczego, pojedyncze urzÄ…dzenie klienckie bÄ™dzie w sposĂłb sekwencyjny wysyĹ‚aÄ‡ ĹĽÄ…dania do zdefiniowanego zbioru punktĂłw koĹ„cowych. Interfejsy programowania aplikacji, ktĂłre implementowaÄ‡ bÄ™dÄ… okreĹ›lone mechanizmy pamiÄ™ci podrÄ™cznej, zwracaÄ‡ bÄ™dÄ… odpowiedĹş na ĹĽÄ…danie, ktĂłrej wartoĹ›Ä‡ rĂłĹĽniÄ‡ bÄ™dzie siÄ™ w sposĂłb znaczny, w zaleĹĽnoĹ›ci od tego, czy API musiaĹ‚o odwoĹ‚aÄ‡ siÄ™ do systemu bazodanowego, czy teĹĽ pozyskaÄ‡ dane z cache. Pierwszy z systemĂłw pamiÄ™ci podrÄ™cznej uwzglÄ™dniaÄ‡ bÄ™dzie staĹ‚y czas waĹĽnoĹ›ci wpisu, natomiast drugi system (tj. system autorski) wyliczaÄ‡ bÄ™dzie czas waĹĽnoĹ›ci na podstawie czÄ™stotliwoĹ›ci odwoĹ‚aĹ„ do punktu koĹ„cowego. Zebrane rezultaty rozpatrywane bÄ™dÄ… dla rĂłĹĽnych przedziaĹ‚Ăłw czasowych. W kaĹĽdym z przedziaĹ‚Ăłw, wydajnoĹ›Ä‡ systemu pamiÄ™ci podrÄ™cznej determinowaÄ‡ bÄ™dzie Ĺ›redni czas odpowiedzi na ĹĽÄ…danie oraz liczba odwoĹ‚aĹ„ do systemu bazodanowego. Badanie zostanie powtĂłrzone w obrÄ™bie obu porĂłwnywanych technologii programistycznych. Po zgromadzeniu wynikĂłw badaĹ„ przeprowadzone zostanÄ… parowe testy statystyczne wykazujÄ…ce istotnoĹ›Ä‡ rĂłĹĽnic pomiarowych.

Kryteria porĂłwnawcze:

- Ĺšredni czas odpowiedzi interfejsu programowania aplikacji na ĹĽÄ…danie klienta w okreĹ›lonym przedziale czasowym.
- Liczba odwoĹ‚aĹ„ interfejsu programowania aplikacji do systemu bazodanowego w okreĹ›lonym przedziale czasowym.

Parametry badania:

- DĹ‚ugoĹ›Ä‡ przedziaĹ‚u czasowego.
- Technologia programistyczna zastosowana do implementacji interfejsu programowania aplikacji.
- Rodzaj zaimplementowanego mechanizmu przechowywania danych w pamiÄ™ci podrÄ™cznej.

[c]|llll| Scenariusz badawczy - badanie przeprowadzone w kontekĹ›cie wdraĹĽania oprogramowania na platformach chmurowych

Nazwa scenariusza badawczego:	

ZmiennoĹ›Ä‡ wydajnoĹ›ci interfejsu API wdroĹĽonego na generycznej oraz dedykowanej platformie chmurowej

Topologia fizyczna Ĺ›rodowiska badawczego:	

Konfiguracja pierwsza rozproszonego Ĺ›rodowiska badawczego **??**

CzynnoĹ›ci implementacyjne:	

- Zaimplementowanie interfejsĂłw programowania aplikacji w technologiach C#/.NET oraz NodeJS/ExpressJS.
- Konfiguracja interfejsĂłw programowania aplikacji w celu obsĹ‚ugi dedykowanych systemĂłw bazodanowych (rodzaj systemu bazodanowego bÄ™dzie zaleĹĽny od wynikĂłw badaĹ„ uzyskanych poprzez realizacjÄ™ scenariusza badawczego **??**).
- WdroĹĽenie interfejsĂłw programowania aplikacji na wirtualnych serwerach prywatnych
- WdroĹĽenie interfejsĂłw programowania aplikacji na dedykowanych wzglÄ™dem okreĹ›lonej technologii platformach chmurowych
- Implementacja mechanizmĂłw pomiaru czasu wykonywanych operacji wewnÄ…trz interfejsĂłw API
- Konfiguracja topologii fizycznej rozproszonego Ĺ›rodowiska badawczego.
- Konfiguracja narzÄ™dzia do wykonywania testĂłw wydajnoĹ›ciowych.

CzynnoĹ›ci badawcze:	

- Realizacja testĂłw wydajnoĹ›ciowych z uwzglÄ™dnieniem zmiennej liczby klientĂłw API.
- Obserwacja oraz gromadzenie wartoĹ›ci pomiarowych dotyczÄ…cych kryteriĂłw porĂłwnawczych.
- Dostosowywanie wartoĹ›ci parametrĂłw przeprowadzanego badania.
- Analiza statystyczna otrzymanych wynikĂłw.

Warunki poczÄ…tkowe podjÄ™cia czynnoĹ›ci badawczych:	

Przed realizacjÄ… testĂłw wydajnoĹ›ciowych zagwarantowana zostanie poprawnoĹ›Ä‡ dziaĹ‚ania kaĹĽdego z interfejsĂłw programowania aplikacji poprzez wykonanie ewaluacji funkcjonalnej. Ponadto, zweryfikowana zostanie dostÄ™pnoĹ›Ä‡ kaĹĽdej z platform chmurowych w czasie wykonywania testĂłw.

Opis scenariusza badawczego:

Po wykonaniu konfiguracji topologii fizycznej rozproszonego Ĺ›rodowiska badawczego, urzÄ…dzenia klienckie bÄ™dÄ… rĂłwnolegle generowaĹ‚y ĹĽÄ…dania hipertekstowe w kierunku interfejsu programowania aplikacji. PoszczegĂłlny interfejs API, analogicznie do obsĹ‚ugiwanego przez niego systemu bazodanowego, wdroĹĽony zostanie w okreĹ›lonym Ĺ›rodowisku chmurowym. Od momentu uzyskania ĹĽÄ…dania od aplikacji klienckiej, zliczany bÄ™dzie czas wykonywania operacji wewnÄ…trz API. Punktem koĹ„cowym czasu realizacji obliczeĹ„ bÄ™dzie chwila wygenerowania odpowiedzi na ĹĽÄ…danie. Czas odpowiedzi liczony w ten sposĂłb, bÄ™dzie odpowiednio pomniejszony wzglÄ™dem standardowego czasu odpowiedzi na ĹĽÄ…danie i nie bÄ™dzie on uwzglÄ™dniaĹ‚, faktu dostarczenia oraz zwrĂłcenia wiadomoĹ›ci http w ramach sieci rozlegĹ‚ej. Badanie zostanie powtĂłrzone w obrÄ™bie obu porĂłwnywanych technologii programistycznych, uwzglÄ™dniajÄ…c dwa systemy bazodanowe, dla ktĂłrych efektywnoĹ›Ä‡ wynikajÄ…ca ze scenariusza badawczego **??** okazaĹ‚a siÄ™ najwyĹĽsza. UwzglÄ™dniona zostanie ponadto zmienna liczba rĂłwnolegle generowanych ĹĽÄ…daĹ„. Po zgromadzeniu wynikĂłw badaĹ„ przeprowadzone zostanÄ… parowe testy statystyczne wykazujÄ…ce istotnoĹ›Ä‡ rĂłĹĽnic pomiarowych.

Kryteria porĂłwnawcze:

- Czas obsĹ‚ugi ĹĽÄ…dania wewnÄ…trz interfejsu programowania aplikacji.

Parametry badania:

- Liczba klientĂłw rĂłwnolegle generujÄ…cych ĹĽÄ…dania.
- Wykorzystywana platforma chmurowa oraz wirtualny serwer prywatny.

Rozdział 4

Redakcja pracy

4.1. UkĹ‚ad pracy

Standardowo praca powinna byÄ‡ zredagowana w nastÄ™pujÄ…cym ukĹ‚adzie:

Strona tytuĹ‚owa
Strona z dedykacjÄ… (opcjonalna)
Spis treĹ›ci
Spis rysunkÓw (opcjonalny)
Spis tabel (opcjonalny)
SkrÓty (wykaz opcjonalny)
1. WstÄ™p
  1.1 Cel i zakres pracy
  1.2 UkĹ‚ad pracy
2. Kolejny rozdziaĹ‚
  2.1 Sekcja
    2.1.1 Podsekcja
      Nienumerowana podpodsekcja
        Paragraf
...
#. Podsumownie i wnioski
Literatura
A. Dodatek
  A.1 Sekcja w dodatku
...
$. ZawartoĹ›Ä‡ pĹ‚yty CD/DVD
Indeks rzeczowy (opcjonalny)

Spis treĹ›ci – powinien byÄ‡ generowany automatycznie, z podaniem tytuĹ‚Ăłw i numerĂłw stron. Typ czcionki oraz wielkoĹ›Ä‡ liter spisu treĹ›ci powinny byÄ‡ takie same jak w niniejszym wzorcu.

Spis rysunkÓw, Spis tabel – powinny byÄ‡ generowane automatycznie (podobnie jak Spis treĹ›ci). Elementy te sÄ… opcjonalne (robienie osobnego spisu, w ktÓrym na przykĹ‚ad sÄ… tylko dwie pozycje specjalnie nie ma sensu).

WstÄ™p – pierwszy rozdziaĹ‚, w ktÓrym powinien znaleĹşÄ‡ siÄ™ opis dziedziny, w jakiej osadzona jest praca, oraz wyjaĹ›nienie motywacji do podjÄ™cia tematu. W sekcji „Cel i zakres" powinien znaleĹşÄ‡ siÄ™ opis celu oraz zadaĹ„ do wykonania, zaĹ› w sekcji „UkĹ‚ad pracy" – opis zawartoĹ›ci kolejnych rozdziaĹ‚Ăłw.

Podsumowanie – w rozdziale tym powinny byÄ‡ zamieszczone: podsumowanie uzyskanych efektÄłw oraz wnioski koĹ„cowe wynikajÄ…ce z realizacji celu pracy dyplomowej.

Literatura – wykaz ĹşrÄłdeĹ‚ wykorzystanych w pracy (do kaĹ‚dego ĹşrÄłdĹ‚a musi istnieÄ‡ odpowiednie cytowanie w tekĹ›cie). Wykaz ten powinien byÄ‡ generowany automatycznie.

Dodatki – miejsce na zamieszczanie informacji dodatkowych, jak: Instrukcja wdroĹ‚eniowa, Instrukcja uruchomieniowa, PodrÄ™cznik uĹ‚ytkownika itp. Osobny dodatek powinien byÄ‡ przeznaczony na opis zawartoĹ›ci doĹ‚Ä…czonej pĹ‚yty CD/DVD. ZaĹ‚oĹ‚ono, Ĺ‚e bÄ™dzie to zawsze ostatni dodatek.

Indeks rzeczowy – miejsce na zamieszczenie kluczowych wyrazÄłw, do ktÄłrych czytelnik bÄ™dzie chciaĹ‚ siÄ™gnÄ…Ä‡. Indeks powinien byÄ‡ generowany automatycznie. Jego zaĹ‚Ä…czanie jest opcjonalne.

4.2. Styl

Zasady pisania pracy (przy okazji moĹ‚na tu zaobserwowaÄ‡ efekt wyrÄłwnania wpisÄłw wystÄ™pujÄ…cych na liĹ›cie wyliczeniowej uzaleĹ‚nione od dĹ‚ugoĹ›ci etykiety):

1. Praca dyplomowa powinna byÄ‡ napisana w formie bezosobowej („w pracy pokazano ...”). Taki styl przyjÄ™to na uczelniach w naszym kraju, choÄ‡ w krajach anglosaskich preferuje siÄ™ redagowanie treĹ›ci w pierwszej osobie.
2. W tekĹ›cie pracy moĹ‚na odwoĹ‚aÄ‡ siÄ™ do myĹ›li autora, ale nie w pierwszej osobie, tylko poprzez wyraĹ‚enia typu: „autor wykazaĹ‚, Ĺ‚e ...”.
3. OdwoĹ‚ujÄ…c siÄ™ do rysunkÄłw i tabel naleĹ‚y uĹ‚ywaÄ‡ zwrotÄłw typu: „na rysunku pokazano ...”, „w tabeli zamieszczono ...” (tabela i rysunek to twory nieĹ‚ywotne, wiÄ™c „rysunek pokazuje” jest niepoprawnym zwrotem).
4. Praca powinna byÄ‡ napisana jÄ™zykiem formalnym, bez wyraĹ‚eĹ„ Ĺ‚argonowych („sejwowanie” i „downloadowanie”), nieformalnych czy zbyt ozdobnych („najznamienitszym przykĹ‚adem tego niebywaĹ‚ego postÄ™pu ...”)
5. PiszÄ…c pracÄ™ naleĹ‚y dbaÄ‡ o poprawnoĹ›Ä‡ stylistycznÄ… wypowiedzi
   - trzeba pamiÄ™taÄ‡, do czego stosuje siÄ™ „liczba”, a do czego „iloĹ›Ä‡”,
   - nie „szereg funkcji” tylko „wiele funkcji”,
   - redagowane zdania nie powinny byÄ‡ zbyt dĹ‚ugie (lepiej podzieliÄ‡ zdanie wielokrotnie zĹ‚oĹ‚one na pojedyncze zdania),
   - itp.
6. ZawartoĹ›Ä‡ rozdziaĹ‚Ăłw powinna byÄ‡ dobrze wywaĹ‚ona. Nie wolno wiÄ™c generowaÄ‡ sekcji i podsekcji, ktÄłre majÄ… zbyt maĹ‚o tekstu lub znaczÄ…co rÄłĹ‚niÄ… siÄ™ objÄ™toĹ›ciÄ…. Zbyt krÄłtkie podrozdziaĹ‚y moĹ‚na zaobserwowaÄ‡ w przykĹ‚adowym rozdziale **??**.
7. Niedopuszczalne jest pozostawienie w pracy bĹ‚Ä™dÄłw ortograficznych czy tzw. literÄłwek – moĹ‚na je przecieĹ‚ znaleĹşÄ‡ i skorygowaÄ‡ automatycznie.

10005. Niedopuszczalne jest pozostawienie w pracy bĹ‚Ä™dÄłw ortograficznych czy tzw. literÄłwek – moĹ‚na je przecieĹ‚ znaleĹşÄ‡ i skorygowaÄ‡ automatycznie.

Rozdział 5

Uwagi techniczne

5.1. Rysunki

W niniejszym szablonie numeracja rysunkw odbywa si automatycznie wedug nastpujcych regu: rysunki powinny mie numeracj cig w obrbie danego rozdziau, sam za numer powinien skada si z dwch liczb rozdzielonych kropk. Pierwsza liczb ma by numer rozdziau, drug – kolejny numer rysunku w rozdziale. Przykadowo: pierwszy rysunek w rozdziale 1 powinien mie numer 1.1, drugi – numer 1.2 itd., pierwszy rysunek w rozdziale 2 powinien mie numer 2.1, drugi – numer 1.2 itd.

Rysunki powinny by wyrodkowane na stronie wraz z podpisem umieszczonym na dole. Podpisy nie powinny koczy si kropk. Czcionka podpisu powinna by mniejsza od czcionki tekstu wiodcego o 1 lub 2 pkt (w szablonie jest to czcionka rozmiaru `small`). Ponadto naley zachowywa odpowiedni odstp midzy rysunkiem, podpisem rysunku a tekstem rozdziau. W przypadku korzystania z szablon odstpy te regulowane s automatycznie. Podpis i grafika musz stanowi jeden obiekt. Chodzi o to, e w edytorach tekstu typu Office podpis nie scala si z grafik i czasem trafia na nastpn stron, osieracajc grafik. Korzystajcym z niniejszego szablonu i otoczenia `\figure` takie osierocenie nigdy si nie zdarzy.

Do kadego rysunku musi istnie odwoanie w tekcie (inaczej mwic: niedopuszczalne jest wstawienie do pracy rysunku bez opisu). Odwoania do rysunkw powinny mie posta: „Na rysunku 3.3 przedstawiono...” lub „... co ujto na odpowiednim schemacie (rys. 1.7)”. Jeli odwoanie stanowi cz zdania, to wtedy wyraz „rysunek” powinien pojawi si w caoci. Jeli za odwoanie jest ujte w nawias (jak w przykadzie), wtedy naley zastosowa skrt „rys.”. Jeli do stworzenia obrazka wykorzystano jakie rda, to powinny one by zacytowane w podpisie tego rysunku.

Naley pamita o tym, e „rysunki” to twory nieywotne. W zwizku z tym nie mog ”pokazywa”. Dlatego „rysunek 1.1 pokazuje ...” jest stylistycznie niepoprawne. Zamiast tego zwrotu trzeba uy „ na rysunku 1.1 pokazano ...”.

Rysunki mona wstawia do pracy uywajc polecenia `\includegraphics`. Zalecane jest, aby pliki z grafikami byy umieszczane w katalogach odpowiadajcych numerom rozdziaw czy literom dodatkw: `rys01`, `rysA` itd. Sposb wstawiania rysunkw do pracy zademonstrowano na przykadze rysunkw **??** i **??**.

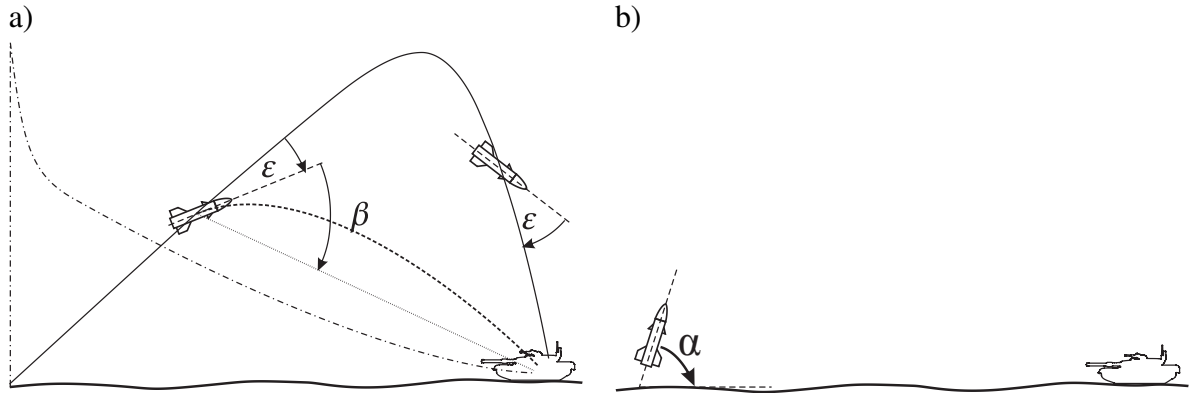
Listing 5.1: Kod rdowy przykadw wstawiania rysunkw do pracy

```
\begin{figure}[ht]
 \centering
  \includegraphics[width=0.3\linewidth]{rys05/kanji-giri}
 \caption{Dwa znaki kanji - giri}
 \label{fig:kanji-giri}
\end{figure}
```

```
\begin{figure}[htb]
 \centering
  \begin{tabular}{@{}ll@{}}
  a) & b) \\
  \includegraphics[width=0.475\textwidth]{rys05/alfa1} &
  \includegraphics[width=0.475\textwidth]{rys05/beta1}
        % jeli obraki s rnej wysokoci, mona je wyrwna do gry stosujc vtop
            ↪ jak niej
        % \vtop{\vskip-2ex\hbox{{\includegraphics[width=0.475\textwidth]{
            ↪ rys05/beta1}}}} &
        % \vtop{\vskip-2ex\hbox{{\includegraphics[width=0.475\textwidth]{
            ↪ rys05/alfa1}}}}
  \end{tabular}
 \caption{Wyznaczanie trajektorii lotu rakiety:
 a) trzy podejcia, b) podejcie praktyczne}
 \label{fig:alfabeta}
\end{figure}
```

義理

Rys. 5.1: Dwa znaki kanji – giri



Rys. 5.2: Wyznaczanie trajektorii lotu rakiety: a) trzy podejcia, b) podejcie praktyczne

Grafiki wektorowe powinny by dostarczone w plikach o formacie pdf. Rozmiar strony w pliku pdf powinien by troszeczk wikszy ni zamieszczona na nim grafika (prosz spojrze na przykady grafik wykorzystanych w niniejszym szablonie). Chodzi o to, aby na rysunku nie pojawiaa si niepotrzebna biaa przestrze. Grafiki rastrowe (gwnie zrzuty z ekranu bd zdjcia) powinny by dostarczane w plikach o formacie png z kompresj bezstratn. Zastosowanie kompresji stratnej, jak jpg, wprowadza niepotrzebne artefakty. Podobnie jak w przypadku grafik wektorowych, grafiki rastrowe nie powinny mie biaych marginesw.

Na rysunkach nie powinno stosowa si 100% czarnego wypenienia, bo robi si plamy przebijajce si przez kartk. Zamiast tego wypenienie powinno by ok. 90% czerni.

Czcionka na rysunkach nie moe by wiksza od czcionki wiodcej tekstu (jedyny wyjtek to np. jakie nagwki). Naley stosowa czcionk kroju Arial, Helvetica bd tego samego kroju co czcionka dokumentu (`texgyre-termes`).

Jeli na jednym rysunku pojawi si ma kilka grafik, to zamiast stosowa `subfigure` lub inne otoczenia naley wstawi grafiki w tabel, opisa j indeksami a) i b), a potem odnie si do tego w podpisie (rys. **??**). Czasem pomaga w pozycjonowaniu rysunkw uycie komendy: `\vtop{\vskip3ex\hbox{\includegraphics[width=0.475\textwidth]{nazwa}}}`

Na rysunkach nie wolno naduywa kolorw oraz ozdobnikw (wiele narzdzi do tworzenia diagramw dostarcza grafik z cieniowaniem, gradacj kolorw itp. co niekoniecznie przekada si na czytelno rysunku).

Podczas rozbienia zrzutw z ekranu naley zadba o to, by taki zrzut by czytelny po wydrukowaniu. Czyli aby pojawiajce si literki byy wystarczajco due, a przestrzenie bez treci – relatywnie mae. Przystpujc do robienia zrzutu trzeba odpowiednio wyskalowa elementy na ekranie. Na przykad robic zrzut z przegldarki FF najpierw naley wcisn CTR–0 (domylne skalowanie), potem CTR–- (zmniejszenie skali o stopie). Potem dobrze jest zawzi okno przegldarki tak, by interesujca tre wypenia je w caoci. Jeli na obserwowanej stronie jest zbyt duo pustych obszarw, to naley je jako zawzi (sterujc wielkoci okna przegldarki lub aktywnymi elementami interfejsu uytkownika). Zrzut bowiem wcale nie musi by odzwierciedleniem 1:1 domylnego ukadu obserwowanych elementw. Wane jest, by na zrzucie z ekranu pokaza interesujcy, opisywany fragment i eby ten fragment by czytelny.

Czasem problemem jest tworzenie zrzutw z ekranu, gdy wystpuj na nim dane wraliwe. Istniej dwa sposoby na radzenie sobie z tym problemem. Pierwszy polega na zastpieniu w systemie danych danych rzeczywistych danymi testowymi – wygenerowanymi tylko do celw prezentacji. Zrzut robi si wtedy na bazie danych testowych. Drugi polega na wykonaniu zrzutu z ekranu, na ktrym pokazano dane rzeczywiste, i nastpnie zamianie tych danych ju w pliku graficznym za pomoc odpowiedniego edytora (np. `gimp`). Czyli oryginalny zrzut z ekranu naley otworzy w edytorze, a potem nadpisa oryginalny tekst wasnym tekstem. Konieczne jest wtedy dobranie odpowiednich czcionek aby nie byo wida wprowadzonych zmian.

> Uwaga: takie manipulowanie zrzutami jest usprawiedliwione jedynie w przypadku koniecznoci ochrony danych wraliwych czy te lepszego pokazania wybranych elementw. Nie moe to prowadzi generowania faszywych rezultatw!!!

5.2. Wstawianie kodu rdowego

Kod rdowy mona wstawia jako blok tekstu pisany czcionk maszynow. Uywa si do tego otoczenie `\lstlisting`. W atrybutach otoczenia mona zdefiniowa tekst podpisu wstawianego wraz z numerem nad blokiem, etykiet do tworzenia odwoa, sposb formatowania i inne ustawienia. Zaleca si stosowanie w tym otoczeniu nastpujcych parametrw:

```
\begin{lstlisting}[label=list:req1,caption=Initial HTTP Request,
                   basicstyle=\footnotesize\ttfamily]
```

Szczeglnie przydatne podczas wstawiania wikszej iloci kodu rdowego jest zastosowanie parametru `basicstyle=\footnotesize\ttfamily`. Dziki niemu zmniejsza si czcionka, a przez to na stronie mona zmieci dusze linijki kodu. Uycie tak zdefiniowanego parametru nie jest jednak sztywnym zaleceniem. Wielko czcionki mona dobiera do potrzeb.

Listing 5.2: Initial HTTP Request

```
GET /script/Articles/Latest.aspx HTTP/1.1
Host: www.codeproject.com
Connection: keep-alive
Cache-Control: max-age=0
Accept: text/html,application/xhtml+xml,application/xml
User-Agent: Mozilla/5.0 ...
Accept-Encoding: gzip,deflate,sdch
```

```
Accept-Language: en-US...
Accept-Charset: windows-1251,utf-8...
```

Mona te sformatowa kod bez stosowania numerowanego podpisu (wtedy nie zamieszcza si caption na licie atrybutw).

```
GET /script/Articles/Latest.aspx HTTP/1.1
Host: www.codeproject.com
Connection: keep-alive
Cache-Control: max-age=0
Accept: text/html,application/xhtml+xml,application/xml
User-Agent: Mozilla/5.0 ...
Accept-Encoding: gzip,deflate,sdch
Accept-Language: en-US...
Accept-Charset: windows-1251,utf-8...
```

Istnieje moliwo wstawiania kodu rdowego w biecej linijce tekstu. Mona to zrobi na kilka sposobw:

- korzystajc z polecenia \texttt ustawiajcego czcionk maszynow, jak w przykadzie tutaj (efekt zastosowania komendy \texttt{tutaj}). Problemem jednak mog okaza si znaki podkrelenia i inne znaki kontrolne.
- korzystaj z otoczenia \verb zapewniajcego wypisanie kodu czcionk maszynow jak w przykadzie tutaj (efekt zastosowania komendy \verb|tutaj|). Problemem jest to, e polecenie \verb nie potrafi ama duszego tekstu.
- korzystajc z polecenia \lstin umoliwiajcego wypisanie kodu czcionk ustawian w opcjach jak w przykadzie tutaj (efekt komendy \lstset{basicstyle=\ttfamily}\lstinline{tutaj}) lub tutaj (efekt komendy \lstinline[basicstyle=\ttfamily]=tutaj=).

5.3. Wykaz literatury oraz cytowania

Cytowania powinny by zamieszczane w tekcie z uyciem komendy \cite{}. Jej argumentem powinien by klucz cytowanej pozycji (lub lista kluczy rozdzielonych przecinkiem bez spacji, jeli takich pozycji w danym miejscu cytuje si wicej) jaki jest uywany w bazie danych bibliograficznych (plik dokumentacja.bib). Po kompilacji bibtex i pdflatex w tekcie pojawia si waciwy odsyacz do pozycji w wykazie literatury (ujty w kwadratowe nawiasy – zgodnie z tym, co definiuje styl plabbrv.bst), za w samym wykazie (rozdzia Literatura) – zacytowana pozycja. Przykadem cytowania jest: „dobrze to opisano w pracach [**?**, **?**]" (gdzie zastosowano komend \cite{JS07,SQL2}).

Co do zawartoci rekordw bibliograficznych - style bibtexowe potrafi „skraca" imiona (czyli wstawia, jeli taka wola, inicjay zamiast penych imion). Niemniej dobrze jest od razu przyj jak konwencj. Proponuje si, aby w rekordach od razu wstawiane byy inicjay zamiast penych imion.

Niekiedy tytuy prac zawieraj wyrazy z duymi i maymi literami. Takie tytuy naley bra w podwjne nawiasy klamrowe, aby bibtex nie zamieni ich na posta, w ktrej poza pierwsz liter pozostae s mae.

Jeli jaki cytowany zasb pochodzi z Internetu, to jego rekord w pliku bib powinien wyglda jak niej.

```
@INPROCEEDINGS{SQL2,
  title={{A MySQL-based data archiver: preliminary results}},
  author={Bickley, M. and Slominski, Ch.},
  booktitle = {{Proceedings of ICALEPCS07}},
        month = oct,
        day = {15--19},
```

```
        year={2007},
  note={\url{http://www.osti.gov/scitech/servlets/purl/922267}
        [dostp dnia 20 czerwca 2015]}
}
```

A to inny przykad rekordu danych bibliograficznych:

```
@TechReport{JS07,
        author = {Jdrzejczyk, J. and rdka, B.},
        title  ={Segmentacja obrazw metod drzew decyzyjnych},
        year = {2007},
        institution = {Politechnika Wrocawska, Wydzia Elektroniki}
}
```

5.4. Indeks rzeczowy

Generowanie indeksu po trosze wyglda jak generowanie wykazu literatury – wymaga kilku krokw. Podczas pierwszej kompilacji `pdflatex` generowany jest plik z rozszerzeniem `*.idx` (zawierajcy „surowy indeks"). Nastpnie, bazujc na tym pliku, generowany jest plik z rozszerzeniem `*.ind` zawierajcy sformatowane dane. Ten krok wymaga uruchomienia odpowiedniego narzdzia oraz zastosowania plik z definicj stylu `Dyplom.ist`. W kroku ostatnim dokonuje si kolejnej kompilacji `pdflatex` (dziki niej w wynikowym dokumencie pojawi si Indeks rzeczowy). Domylnie Indeks rzeczowy zostanie sformatowany w ukadzie dwukolumnowym.

Oczywicie aby to wszystko zadziaao w kodzie szablonu naley umieci odpowiednie komendy definiujce elementy indeksu rzeczowego (`\index`) oraz wstawiajce sformatowany Indeks rzeczowy do dokumentu wynikowego (`\printindex`). Wicej informacji o tworzeniu indeksu rzeczowego mona znale na stronie `https://en.wikibooks.org/wiki/LaTeX/Indexing`. Poniej przedstawiono przykady komend uytych w szablonie do zdefiniowania elementw indeksu rzeczowego:

- `\index{linia komend}` – pozycji gwna.
- `\index{generowanie!-- indeksu}` – podpozycja.

Generowanie pliku `*.ind` mona inicjowa na kilka sposobw:

- poprzez wydanie odpowiedniego polecenia bezporednio w linii komend
  ```
  makeindex Dyplom.idx -t Dyplom.ilg -o Dyplom.ind -s Dyplom.ist
  ```
- poprzez odpalenie odpowiedniego narzdzia rodowiska. Na przykad w `TeXnicCenter` definiuje si tzw. `output profiles`:
  ```
  makeindex "%tm.idx" -t "%tm.ilg" -o "%tm.ind" -s "%tm.ist"
  ```
  a samo generowanie pliku `*.ind` zapewni wybranie pozycji menu `Build/Makeindex`.
- korzystajc z odpowiednio sparametryzowanych pakietw i komend wewntrz kompilowanego dokumentu (czyli od razu przy okazji jego kompilacji).
  ```
  \DisemulatePackage{imakeidx}
  \usepackage[noautomatic]{imakeidx}
  % jeli chcemy, by indeks by generowany automatycznie programem makeindex:
  %\usepackage[makeindex]{imakeidx}
  % a tak pono mona przekaza opcje do programu generujcego indeks:
  %\makeindex[options=-s podrecznik -L polish -M lang/polish/utf8]
  %\makeindex[options=-s podrecznik]
  \makeindex
  ```
  Niestety, `makeindex` jest narzdziem, ktre umieszcza cz pozycji w grupie `Symbols`, a nie w grupach zwizanych z literkami alfabetu. W zwizku z czym indeksowany element zaczynajcy si od polskiej literki trafia do grupy `Symbols`, jak np. `\index{wiato}`. Jeli chce

si zamieszcza w indeksie symbole matematyczne, to dobrze jest to robi jak w nastpujcym przykadzie: `\index{$asterisk@$\ast$}` czy te `\index{c@$\mathcal{C}$}`, tj. dostarczajc przy okazji klucz do sortowania. Lepiej w tym wzgldzie radz sobie inne narzdzia, jak `texindy` lub `xindy` dostpne pod linuxem. Korzystajc z nich uzyskuje si grupy polskich literek w indeksie rzeczowym (hasa zaczynajce si od polskich literek ju nie trafiaj do grupy Symbols). Przykad polecenia wydanego z linii komend, w ktrym wykorzystano `texindy` zamieszczono poniej (zakadamy kodowanie plikw w UTF8, mona dla niniejszego szablonu zmieni na cp1250):

```
texindy -L polish -M lang/polish/utf8 Dyplom.idx
```

To polecenie wygeneruje `Dyplom.ind` o zawartoci:

```
\begin{theindex}
  \providecommand*\lettergroupDefault[1]{}
  \providecommand*\lettergroup[1]{%
      \par\textbf{#1}\par
      \nopagebreak
  }

  \lettergroup{G}
  \item generowanie
    \subitem -- indeksu, 27
    \subitem -- wykazu literatury, 27

  \indexspace

  \lettergroup{L}
  \item linia komend, 27

  \indexspace

  \lettergroup{}
  \item \'Swiat\IeC {\l }o, 28

\end{theindex}
```

Aby mie wiksz kontrol automatyczne generowanie indeksu zostao w niniejszym szablonie wyczone (indeks trzeba wygenerowa samemu, wydajc polecenie `makeindex` lub zalecane `texindy`).

5.5. Inne uwagi

Dobrym sposobem na kontrol bdw wystpujcych podczas kompilacji jest wstawianie linijki `\end{document}` w wybranym miejscu dokumentu. Jest to szczeglnie przydatne w przypadkach, gdy bdy te s trudne do zidentyfikowania (gdy wygenerowane przez kompilator numery linii z bdami nie s tymi, w ktrych bdy wystpuj). Wystarczy wtedy przestawi wspomnian linijk do kolejnych miejsc, a znajduj to miejsce, gdzie wystpuje problem.

Aby osign apostrofy maszynowe (czyli takie zoone z samych kresek) naley uy polecenia `"{}jak tutaj{}"` (podwjny apostrof i podwjny apostrof z na wszelki wypadek umieszczonymi nawiasami klamrowymi, nawiasy s potrzebne z tej racji, i podwjny apostrof przed niektrymi literkami zamienia je na literki z akcentami). W efekcie otrzymamy "jak tutaj". Jeli natomiast apostrofy maj by drukarskie (czyli zoone z kropek i kresek), to naley uy polecenia `,,jak tutaj''` (dwa pojedyncze przecinki i dwa pojedyncze apostrofy). W efekcie otrzymamy „jak tutaj”. Mona te uy znakw apostrofw odpowiednio zakodowanych jak tutaj, tylko e czasem trudno pisze si takie apostrofy w rodowiskach kompilacji projektw latexowych.

Oto sposoby ustawienia odstpw midzy liniami:

- uywajc komendy `\linespread{...}` (akceptowalne), przy czym atrybutem tej metody jest wspczynnik zaleny od wielkoci czcionki. Dla czcionki wiodcej 12pt odstp ptora linii osignie si komend `\linespread{1.241}`. Dla innych czcionek wiodcych wartoci tego parametru s jak w poniszym zestawieniu.

```
10pt 1.25 dla \onehalfspacing
     1.667 for \doublespacing,
                 poniewa ,,basic ratio'' = 1.2
                (\normalfont posiada \baselineskip rozmiaru 12pt)
11pt 1.213 dla \onehalfspacing oraz 1.618 dla \doublespacing,
     poniewa ,,basic ratio'' = 1.236
                (\normalfont posiada \baselineskip rozmiaru 13.6pt)
12pt 1.241 dla \onehalfspacing oraz 1.655 dla \doublespacing,
     poniewasince ''basic ratio'' is 1.208
                (\normalfont has a \baselineskip of 14.5pt)
```

  Kopot w tym, e raz ustawiony odstp bdzie obowizywa do wszystkich czcionek (nie dziaa tu adem mechanizm zmiany wspczynnika w zalenoci od wielkoci czcionki akapitu).
- uywajc pakietu `setspace` (niezalecane). Poniewa klasa `memoir` emuluje pakiet `setspace`, w preambule dokumentu naleaoby umieci:

```
\DisemulatePackage{setspace}
\usepackage{setspace}
```

  a potem mona ju sterowa odstp komendami:

```
\singlespacing
\onehalfspacing
\doubelspacing
```

  Ten sposb pozwala na korzystanie z mechanizmu automatycznej zmiany odlegoci linii w zalenoci od wielkoci czcionki danego akapitu.
- korzystajc bezporednio z komend dostarczonych w klasie `memoir` (zalecane):

```
\SingleSpacing
\OnehalfSpacing
\DoubleSpacing
```

  Ten sposb rwnie pozwala na korzystanie z mechanizmu automatycznej zmiany odlegoci linii w zalenoci od wielkoci czcionki danego akapitu.

Na koniec jeszcze uwaga o rozmiarze pliku wynikowego. Ot `pdflatex` generuje pliki `pdf`, ktre zazwyczaj mogyby by nieco lepiej skompresowane. Do lepszego skompresowania tych plikw mona uy programu `ghostscript`. Wystarczy w tym celu wyda komend (pod windowsami):

```
gswin64 -sDEVICE=pdfwrite -dCompatibilityLevel=1.4 -dNOPAUSE -dQUIET \
-dSAFER -dBATCH -sOutputFile=Dyplom-compressed.pdf Dyplom.pdf
```

W poleceniu tym mona rwnie wstawi opcj `-dPDFSETTINGS=/prepress` (zapewniajc uzyskanie wysokiej jakoci, zachowanie kolorw, uzyskanie obrazkw w rozdzielczoci 300 dpi). Ze wzgldw licencyjnych ghostscript uywa domylnie algorytmw z kompresj stratn. Przy kompresji moe wic doj do utraty jakoci bitmap.

Rozdział 6

Podsumowanie

Lorem ipsum dolor sit amet eleifend et, congue arcu. Morbi tellus sit amet, massa. Vivamus est id risus. Sed sit amet, libero. Aenean ac ipsum. Mauris vel lectus.

6.1. Sekcja poziomu 1

Lorem ipsum dolor sit amet eleifend et, congue arcu. Morbi tellus sit amet, massa. Vivamus est id risus. Sed sit amet, libero. Aenean ac ipsum. Mauris vel lectus.

Nam id nulla a adipiscing tortor, dictum ut, lobortis urna. Donec non dui. Cras tempus orci ipsum, molestie quis, lacinia varius nunc, rhoncus purus, consectetuer congue risus.

6.1.1. Sekcja poziomu 2

Lorem ipsum dolor sit amet eleifend et, congue arcu. Morbi tellus sit amet, massa. Vivamus est id risus. Sed sit amet, libero. Aenean ac ipsum. Mauris vel lectus.

Sekcja poziomu 3

Lorem ipsum dolor sit amet eleifend et, congue arcu. Morbi tellus sit amet, massa. Vivamus est id risus. Sed sit amet, libero. Aenean ac ipsum. Mauris vel lectus.

Paragraf 4 Lorem ipsum dolor sit amet eleifend et, congue arcu. Morbi tellus sit amet, massa. Vivamus est id risus. Sed sit amet, libero. Aenean ac ipsum. Mauris vel lectus.

6.2. Sekcja poziomu 1

Lorem ipsum dolor sit amet eleifend et, congue arcu. Morbi tellus sit amet, massa. Vivamus est id risus. Sed sit amet, libero. Aenean ac ipsum. Mauris vel lectus.

Dodatek A

Instrukcja wdroeniowa

Jeli praca skoczya si wykonaniem jakiego oprogramowania, to w dodatku powinna pojawi si instrukcja wdroeniowa (o tym jak skompilowa/zainstalowa to oprogramowanie). Przydaoby si rwnie krtkie how to (jak uruchomi system i co w nim zrobi – zademonstrowane na jakim najprostszym przypadku uycia). Mona z tego zrobi osobny dodatek,

Dodatek B

Opis zaczonej pyty CD/DVD

Tutaj jest miejsce na zamieszczenie opisu zawartoci zaczonej pyty. Naley wymieni, co zawiera.