|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ЗМІСТ**  СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ 3  ВСТУП 5  1. АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ТА ОБГРУНТУВАННЯ ТЕМИ ДИПЛОМНОГО ПРОЕКТУ 6  1.1. Опис існуючих рішень інтернет банкінгу 6  1.2. Безпека при виконанні дистанційних платіжних операцій 11  1.2.1 Шифрування у банківських системах 12  1.2.2. Рівень захищених сокетів (SSL) 13  1.2.3. Двофакторний метод аутентифікації (2FA) 14  1.2.4. Використання транзакцій 17  1.3. Обґрунтування теми проекту та обраних засобів розроблення 18  2. АНАЛІЗ ІНСТРУМЕНТІВ РОЗРОБКИ 20  2.1. IntelliJ IDEA 20  2.2. SAP PowerDesigner 21  2.3. MySQL Workbench 22  2.4. MySQL Connector/J 23  2.5. Хеш-фунції SHA-256 23  2.6. Утиліта для ведення журналів Log4j 26  2.7. JSTL 26  3. ОПИС НЕОБХІДНОГО ПРОГРАМНОГО ЗАПЕСПЕЧЕННЯ 28  3.1. Java Virtual Machine 28  3.2. Java cервлет 28 | | | | | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | | | | | |
|  |  |  |  |  |
| Змін. | Арк. | № докум. | Підпис | Дата |
| Розробив | | Горпинич-Радуженко І.О. |  |  | Веб-орієнтована система банківських операцій користувача.  Пояснювальна записка | Літ. | | | Аркуш | Аркушів |
| Перевірив | | Орлова М.М. |  |  |  |  |  | 1 | 50 |
| Консульт. | |  |  |  | НТУУ «КПІ»  Кафедра СПіСКС  Група КВ-41 | | | | |
| Н. контроль | | Клятченко Я.М. |  |  |
| Зав. каф. | | Тарасенко В.П. |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 3.3 JavaServer Page. 29  3.4 Веб-контейнер. 30  3.5. Apache Tomcat 33  3.6. MySQL Server 35  4. АНАЛІЗ НЕОБХІДНОГО АПАРАТНОГО ЗАПЕСПЕЧЕННЯ 37  4.1. Характеристики веб-серверу 37  4.2. Характеристики серверу баз даних 38  4.2.1 Двигун InnoDB 39  4.2.1 RAID10 40  5. РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 41  5.1.Шаблони проектування  5.2. DAO+DTO  5.3.JDBC  5.4.Prepared statements  5.5.Connection Pool  5.6.Services  5.7.Maven  ВИСНОВКИ  СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 2 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ДОДАТКИ**  Додаток 1. Копії графічного матеріалу  Додаток 2. Лістинг програми | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 3 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ**  Об’єкт – сукупність обладнання, яке підключено до інтерфейсного блоку;  SMS (від англ. Short Message Service) – технологія, яка дозволяє здійснювати прийом і передачу коротких текстових повідомлень; | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 4 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ВСТУП**  На сьогоднішній час, дистанційні фінансові операції є невід’ємною частиною соціального існування кожної людини у світі. Цей факт особливо стосується людей малого та середнього фінансового достатку, що є більшою частиною земного населення. При виконанні банківських операцій вже не потрібна особиста присутність людини у банківській установі, достатньо мати ідентифіковані персональні дані у системі банківського контролю та доступ до мережі Інтернет. Є необхідність у підвищенні рівня банківської освіченості, тому що дистанційні банківські операції виконуються у автоматичному режимі.  Банківські операції користувача включають в себе роботу з банківськими онлайн переказами, депозитними та кредитними рахунками, управління банківськими картками.  Банки та інші фінансові установи працюють над розробкою та впровадженням веб-орієнтованих систем через їх потенціал підвищення ефективності, скорочення витрат і залучення нових клієнтів. Споживачі починають використання цих технологій через підвищену зручність та економію часу. У період з 1995 по 2003 рік, кількість електронних банківських операцій збільшилася восьмикратно. У період з кінця 2002 року до початку 2005 року, використання веб-орієнтованих банківських мереж збільшилася на 47 відсотків[1]. Лише в Україні на початок 2017 року системами дистанційних фінансових операцій користуються більш ніж 10 мільйон громадян.[2]  Проте відсутність фінансової освіченості викликає цифровий розрив та віддалення людей малого та середнього фінансового достатку від використання усіх можливих переваг онлайн банкінгу. Тому є необхідність у розробці веб-орієнтованої банківської моделі операцій користувача, для здобуття необхідних навичок використання цих систем. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 5 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **1. АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ТА ОБҐРУНТУВАННЯ**  **ТЕМИ ДИПЛОМНОГО ПРОЕКТУ**  **1.1. Аналіз існуючих систем онлайн банкінгу**  Розглянемо існуючі системи онлайн банкінгу в Україні.  На сьогоднішній день в Україні існує багато веб-орієнтованих систем для надання банківських послуг як від самих банків, так і комерційні рішення інших виробників програмного забеспечення. Розглянемо декілька основних.  Система «Privat24»  Лідером на ринку України є система «Privat24», тому що переважна кількість державного безготівкового грошового обігу виконується за допомогою цього додатку.  Програмний реалізація має необхідну, у сучасний час, дворівневу систему автентифікації .  . Для початку роботи потрібно лише мати телефон, який був указаний при реєстрації у банку. Після декількох годин користувач автоматично буде зареєстрований у системі онлайн банкінгу.    Рис.1.1 Інтерфейс онлайн-банкінгу «Privat24» | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 6 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Система «Ощад24»  Система державного банку - «Ощадбанку». Процедура аутентифікації дуже схожа на «Приват24», тому що потребує підключення SMS банкінгу для можливості дворівневої аутентифікації.  На відміну від програмного рішення «Приватбанку», при переказі грошів між картками власного банку необхідно віконувати лише за номером. При такому підході користувачеві дуже легко помилитися. У системі є базовий функціонал для роботі з депозитними рахунками (можливість роботи лише з декількома видами). Великим недоліком є неможливість використання альтернативних можливостей аутентифікації окрім SMS.  E:\Documents\4 курс\Diploma\o24.jpg  Рис.1.2 Інтерфейс онлайн-банкінгу «Ощад24» | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 7 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Система «UKRSIB online» та «StarAccess»  Один з найголовніших банків України теж запровадив систему дистанційного веб-орієнтованого онлайн банкінгу «UKRSIB online» - для фізичних осіб, «StarAccess» - для юридичних осіб.  C:\Users\John\Downloads\Telegram Desktop\photo_2018-05-20_12-14-23.jpgC:\Users\John\Downloads\Telegram Desktop\photo_2018-05-20_12-14-17.jpgУ 2018 році «Укрсіббанк» зробив ребрендінг свого онлайн банкінгу для фізичних осіб зі «Star24» на «UKRSIB online», і це вдалося їм з великим успіхом. Система побудована на базі новітніх технологій безпеки та багатошарової автентифікації клієнтів. Використовує принцип RESTFul сервісів, тобто з мобільного додатку можна повноцінно користуватись онлайн банкінгом та усіма його функціями. Цікавою особливістю є контроль витрат і надходжень. UKRSIB online в автоматичному режимі проводить аналіз ваших витрат і надходжень, розбиває їх по категоріях та показує в графіках.  Рис.1.3 Інтерфейс мобільного онлайн-банкінгу «UKRSIB online» | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 8 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Рис.1.4 Інтерфейс онлайн-банкінгу «UKRSIB online»  Система **онлайн-банкінгу** StarAccess надає зручний інструмент дистанційного управління коштами для юридичних осіб, відкриваючи перед ними широкий перелік можливостей, високий рівень надійності і гарантією безпеки.  Система передбачає наявність у користувача спеціального брелока з ЄЦП, який підтверджує особистість при роботі з додатком.  Можливо, це є незручністю підключення до системи - обов'язково потрібно відвідати філію банку і отримати ЄЦП. Але система представляє користувачеві стабільну роботу з достатнім набором функцій управління рахунком та високим рівнем безпеки.  Система StarrAccess використовує додаток розроблений на мові Java для забеспечення максимальної надійності виконаних транзакцій. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 9 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| E:\Documents\4 курс\Diploma\s24.jpg  Рис.1.5 Інтерфейс онлайн-банкінгу «StarAccess»  Система онлайн платежів «Portmone»  Система «Portmone» це міжбанківська [електронна платіжна система](https://uk.wikipedia.org/wiki/%D0%95%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BD%D0%BD%D0%B0_%D0%BF%D0%BB%D0%B0%D1%82%D1%96%D0%B6%D0%BD%D0%B0_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0), яка виконує послуги доставки і оплати рахунків за різні товари і послуги в дистанційному режимі через [Інтернет](https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D1%82%D0%B5%D1%80%D0%BD%D0%B5%D1%82), оплата комунальних послуг, переводи з карти на карту (не прив’язуючись до власного банку), поповнення мобільного, отримання грошових переказів за допомогою міжнародних платіжних карток [Visa](https://uk.wikipedia.org/wiki/Visa) і [MasterCard](https://uk.wikipedia.org/wiki/MasterCard). | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 10 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Рис.1.5 Інтерфейс системи онлайн платежів «Portmone»  1.2. Безпека при виконанні дистанційних платіжних операцій  В онлайн-банкінгу, як і в традиційних банківських системах, безпека є основною проблемою. Розробники спеціалізованого програмного забезпечення намагаються вживати всі необхідні запобіжні заходи, щоб переконатися, що інформація клієнтів передається безпечно та надійно. Найновіші методи в безпеці онлайнової банківської системи використовуються для збільшення моніторингу цілісності та безпеки системи. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 11 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Захист онлайн-банкінгу повинен розглядаєтися на трьох рівнях. Перша проблема полягає в безпеці інформації про клієнтів, оскільки вона надсилається з ПК-клієнта на веб-сервер. Друга частина стосується безпеки навколишнього середовища, в якому розташована мережа банківських серверів та база даних клієнтів. Нарешті, вживаються заходи безпеки, щоб запобігти неавторизованим користувачам входити в розділ онлайн-банкінгу веб-сайту.  1.2.1 Шифрування у банківських системах  Захист конфіденційності повідомлень між браузером та серверами банку забезпечується за допомогою шифрування. Шифрування відбувається так: коли користувач переходе на сторінку входу онлайн-банкінгу, браузер встановлює безпечний сеанс на нашому сервері. Безпечний сеанс встановлюється за протоколом Encryption Secure Sockets Layer (SSL).  Цей протокол вимагає обміну, що називаються публічними та приватними ключами. Ключі - випадкові числа, вибрані для цього сеансу, і вони відомі лише між браузером користувача та сервером банку. Після обміну ключами браузер використовуватиме ці цифри для розбиття (шифрування) повідомлень, надісланих від браузера до сервера. Обидві сторони вимагають ключів, оскільки їм потрібно розшифрувати повідомлення, після їх отримання. Протокол SSL не тільки забезпечує конфіденційність, але також гарантує, що жоден інший веб-користувач не зможе "видати себе" за іншого користувача, а також не змінить жодну надіслану інформацію. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 12 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1.2.2. Рівень захищених сокетів (SSL)  Це стандартна технологія безпеки для встановлення зашифрованого зв'язку між сервером і клієнтом, як правило, веб-сервером (веб-сайтом) та веб-переглядачем, або поштовим сервером і поштовим клієнтом (наприклад, Outlook).  SSL дозволяє безпечно передавати конфіденційну інформацію, таку як номери кредитних карток, номери соціального захисту та облікові дані для входу. Зазвичай, дані, що надсилаються між веб-переглядачами та веб-серверами, надсилаються звичайним текстом, залишаючи вас вразливими до прослуховування та крадіжки ваших даних. Якщо зловмисник може перехопити всі дані, що надсилаються між веб-переглядачем та веб-сервером, він може бачити та використовувати цю інформацію.  Більш конкретно, SSL - це протокол безпеки. Протоколи описують, як слід використовувати алгоритми. У цьому випадку протокол SSL визначає змінні шифрування як для прийнятих, так і для переданих даних.  Всі веб-браузери мають можливість взаємодіяти з захищеними веб-серверами за допомогою протоколу SSL. Однак браузеру та серверу потрібно те, що називається SSL-сертифікатом, щоб мати змогу встановити безпечне з'єднання.  SSL захищає мільйони даних людей в Інтернеті щодня, особливо під час онлайн-транзакцій або передачі конфіденційної інформації. Користувач може визначити, чи знаходиться його браузер у беспечному режимі, побачивши символ захисту у вікні веб-браузера.    Рис.1.6 Приклад відображення захищеного режиму | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 13 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1.2.3. Двофакторний метод аутентифікації (2FA)  З кожним днем зростає кількість способів отримання доступу до приватних даних користувача при використанні стандартних процедур безпеки, що вимагають лише просте ім'я користувача та пароль. Шахраї намагаються отримати як особисті та фінансові дані, а потім використовують цю інформацію здійснення шахрайських дій, загалом фінансового характеру.  Двофакторна автентифікація, також відома як 2FA, є додатковим рівнем безпеки, яка вимагає не тільки пароля та імені користувача, але якусь додаткову унікальну інформацію, яку користувач повинен відразу пред’явити – це може буди інформація яка надходить до вашого телефону у вигляді SMS, але це також може бути USB-ключ (апаратний токен) або інший пристрій, який може генерувати одноразові коди.  Користувач, який використовує сервіс з двофакторною, зазвичай надаєте свій номер мобільного телефону (також може використовуватися мобільний додаток, але це найпоширеніший спосіб використання двофакторної автентифікації.). Більшість форм двофакторної автентифікації вимагають входу за допомогою імені користувача та пароля, а потім введення коду, який надіслано вам через SMS. Цей метод не лише підтверджує, що ви знаєте базові данні для авторизації (ім'я користувача та пароль), але і те, до чого має доступ лише користувач (мобільний телефон), номер якого зареєстрований як пристрій для отримання цих кодів.  Використання імені користувача та пароля разом із інформацією, яку знає лише користувач, ускладнює отримання потенційними зловмисникам доступу та викрадення особистих даних користувача. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 14 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Історично склалося так, що двофакторна автентифікація - це не нова концепція, але її використання стало набагато більш поширеним з початку цифрової ери, в якій ми зараз живемо. У лютому 2011 року, Google оголосила про двофакторну автентифікацію для доступну до своїх сервісів, потім до них приєдналися такі гіганти як MSN і Yahoo, Microsoft.  Багато людей, напевно, не знають, що такий тип процесу безпеки називається двофакторною автентифікацією, і, ймовірно, навіть не замислюється про це, коли використовують апаратні токени, виданих їхнім банком для використання з їхньою карткою та особистим ідентифікаційним номером, коли користувач хоче виконати операцію за допомогою онлайн-банкінгу.  Використання процедури аутентифікації за двома факторами може допомогти зменшити кількість випадків крадіжки особистих даних в Інтернеті, а також фішингу через електронну пошту, оскільки злочинцям потрібно більше, ніж просто дані про ім'я користувача та пароль.  Недоліком цього процесу безпеки є те, що нові апаратні токени (у вигляді ключів або кард-рідерів) мають бути замовлені, а потім видані користувачеві, і це може призвести до уповільнення процесу початку роботи з сервісом, щоб отримати доступ до своїх приватних даних за допомогою цієї процедури автентифікації. Токени також зазвичай мають невеликі габарити, тому легко втрачаються, що викликає більше проблем для оформлення нової заяви для виробництва нового ключа. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 16 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| E:\Documents\4 курс\Diploma\photo_2018-05-20_16-39-34.jpg  Рис.1.7 Приклад програмного токену для виконання дворівневої автентифікації у системі онлайн-банкінгу «StarAccess»  1.2.4. Використання транзакцій  Транзакція - це серія операцій, які відокремлюються та виконуютсья як атомарне ціле. Операція у рамках транзакції повинна виконатися або повністю, або повернути дані до початкового стану.  Обробка транзакцій призначена для підтримки цілісності системи (як правило, бази даних або деяких сучасних файлових систем) у відомому стані, забезпечуючи, що взаємозалежні операції в системі або успішно завершені, або всі успішно скасовані. Оновна ідея - незалежно від того, що станеться, (вимкнення електроенергії, катаклізм, або збій у роботі програмного забезпечення), дані, з якими ви працюєте, будуть розглядатися атомарно, та змінюватись атомарно.  Наприклад, розгляньте типову банківську транзакцію, яка передбачає переведення 700 у.о. з ощадного рахунку на дебетовий рахунок клієнта. Ця операція передбачає принаймні дві окремі операції в комп'ютерних умовах: зняття з ощадного рахунку 700 у.о. та поповнення поточного рахунку на 700 у.о.. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 17 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Якщо одна операція успішна, а інша ні, то в кінці дня залишок на картах не будуть співпадати. Отже, існує спосіб забезпечення умов, у яких обидві операції були успішними або обидві невдачі, так щоб в базі даних банку в цілому ніколи не було невідповідності.  Обробка транзакцій пов'язує декілька окремих операцій в єдину, неділиму транзакцію та гарантує, що всі операції в транзакції виконуються без помилок, або жодна з них не виконується. Якщо деякі операції завершені, але виникають помилки, система обробки транзакцій "відкачує" всі операції транзакції (включаючи успішні), тим самим видаляючи всі сліди транзакції та відновлення системи до послідовного початкового стану операції. Якщо всі операції транзакції успішно завершені, операція здійснюється системою, і всі зміни в базі даних стають постійними; транзакцію не можна відкотити після того, коли це буде зроблено.  Обробка транзакцій захищає від апаратних та програмних помилок, які можуть залишити транзакцію частково завершеною. Якщо комп'ютерна система аварійно завершується посередині транзакції, система обробки транзакцій гарантує, що всі операції в будь-яких нездійснених станах скасуються.  1.3. Обґрунтування теми проекту та обраних засобів розроблення  Дана розробка є актуальною, тому що виявлений високий рівень розвитку технологій у сфері дистанційних банківських операцій. Проте відсутність фінансової освіченості викликає цифровий розрив та віддалення людей малого та середнього фінансового достатку від використання усіх можливих переваг онлайн банкінгу. Тому є необхідність у розробці веб-орієнтованої банківської моделі операцій користувача, для здобуття необхідних навичок використання цих систем. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 18 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Для розробки дипломного проекту мною була обрана мова високого рівня Java. Технології мови Java включають у себе великий набір API-інтерфейсів, інструментів і реалізацій часто використовуваних алгоритмів безпеки, механізмів і протоколів. API безпеки Java охоплює широке коло областей, включаючи криптографію, інфраструктуру відкритих ключів, захищений зв'язок, автентифікацію та контроль доступу. Технологія безпеки Java надає розробнику всеохоплюючу основу для написання додатків, а також надає користувачеві або адміністратору набір інструментів для безпечного управління програмами. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 19 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 2. АНАЛІЗ ІНСТРУМЕНТІВ РОЗРОБКИ  2.1. IntelliJ IDEA  IntelliJ IDEA - це інтегроване середовище розробки (IDE) для мови високого рівня Java. Використовується для розробки комп'ютерного програмного забезпечення. Вона пропонує внутрішній аналіз, мультимедійний синтаксис, автодоповнення, історію змін, інтегроване підключення до бази даних, інтегроване рішення для контролю версій, можливість під’єднання серверу для автоматичного завантаження веб-проекту, підключення додатків для підтримки інших мов програмування. IntelliJ IDEA відрізняється від аналогів простотою використання, гнучкістю та надійним дизайном. При використанні IntelliJ IDEA, ви отримаєте інтегроване середовище розробки, але з великою кількістю доповнень, які роблять розробку набагато простішою і приємнішою.Приведемо декілька з них:  * Вкладка автодоповнення (імена класів, функції, методи, змінні) * Чіткіші повідомлення про помилки і їх виділення кольором * Аналіз потоку даних у реальному часі * Гарна інтеграція з модулями (GIT, Maven, TypeScript) * Розумний рефакторінг коду * Виявлення дублікатів   Особливістю IntelliJ IDEA є підтримка [Spring Integration Framework](http://www.springsource.org/spring-integration) - розширення для веб-програмування, що підтримує моделі корпоративної інтеграції.  Недоліком цієї системи є платний доступ для використання. Але кожен студент має можливість зробити підписку для навчальних цілей. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 20 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 2.2. SAP PowerDesigner  SAP PowerDesigner – це програмий додаток для спільного інструментального моделювання архітектури програмного забеспечення та баз даних.  PowerDesigner підтримує дизайн [моделей архітектури](https://en.wikipedia.org/wiki/Model-driven_architecture) програмного забезпечення. PowerDesigner зберігає моделі за допомогою різноманітних розширень файлів. Внутрішня структура файлу може бути або XML, або у форматі, бінарного файлу. PowerDesigner також може зберігати моделі в сховищі бази даних.  PowerDesigner підтримує побудову таких видів моделей, як:   * Модель вимог (RQM). Допомагає вам проаналізувати будь-які письмові вимоги та пов'язати їх * Модель корпоративної архітектури (EAM). Допомагає аналізувати та документувати бізнес-функції * Концептуальна модель (CDM). Допомагає проаналізувати концептуальну структуру інформаційної системи * Об'єктно-орієнтована модель (OOM). Допомагає проаналізувати інформаційну систему через об’єкти використання * Логічна модель даних (LDM). Допомагає проаналізувати структуру інформаційної системи * Фізична модель даних (PDM). Допомагає проаналізувати таблиці, представлення даних та інші об'єкти в базі даних   PowerDesigner має підтримку:   * Створення коду ( Java , C # , Hibernate, JSF, WinForm ) * Моделювання даних (працює з більшістю основних систем RDBMS ) | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 21 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| * Моделювання складу даних ( WarehouseArchitect ) * Моделювання об'єкта ( діаграми UML) * Аналіз вимог * XML моделювання - схеми і DTD стандарти   Недоліком є не можливість неспівпадіння типів даних при побудові архитектури бази даних до конкретно обраної мови програмування.  2.3. MySQL Workbench  MySQL Workbench - це візуальний інструмент розробки баз даних , який інтегрує розробку , адміністрування , створення баз даних , створення та підтримку SQL запитів в єдине інтегроване середовище розробки для бази даних.  MySQL Workbench надає розробникам повний набір візуальних інструментів для створення, редагування та управління SQL запитами, з'єднаннями з базами даних та об'єктами. Візуальний редактор SQL дозволяє розробникам створювати, редагувати та виконувати запити, створювати та редагувати дані, а також переглядати та експортувати результати.  MySQL Workbench надає такі можливості, як:   * Підсвічування синтаксису * контекстно-конфіденційна довідка * автозаповнення * налагодження SQL-інструкцій.   Програма надає набір інструментів для підвищення продуктивності додатків MySQL. DBA можуть швидко переглянути основні показники ефективності, використовуючи інформаційну панель продуктивності. Звіти про ефективність забезпечують легку ідентифікацію високоякісних SQL-операторів, тощо. Крім того, розробники можуть бачити, де оптимізувати свій запит за допомогою покращеного та простого у використанні плану підказок. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 22 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 2.4. MySQL Connector/J  MySQL забезпечує з'єднання для клієнтських додатків, розроблених на мові програмування Java, за допомогою MySQL Connector / J, драйвера, який реалізує API Java Database Connectivity (JDBC) .  2.5. Хеш-фунції SHA-256  SHA-2 ( Secure Hash Algorithm 2 ) - це набір криптографічних хеш-функцій, розроблений Агентством національної безпеки США (NSA). [3] Вони побудовані з використанням структури Меркле - Дамгарда, з функції одностороннього стиснення, яка сама по собі побудована з використанням структури Девіса-Майєра з (класифікованого) спеціалізованого блочного шифру.  Криптографічні хеш-функції - це математичні операції, що виконуються на цифрових даних; шляхом порівняння обчисленого "хеш" числом з відомим і очікуваним значенням хешу, людина може визначити цілісність даних. Наприклад, обчислення хеш завантаженого файлу та порівняння результату з попередньо опублікованим результатом хешу може показати, файл завантажено чи змінено. Ключовим аспектом криптографічних хеш-функцій є їх стійкість до колізій: ніхто не повинен мати змогу знаходити два різних вхідних значення, що призводять до того ж хеш-вихідного сигналу.  SHA-256 - нова хеш-функція, обчислена за допомогою 32-бітових слів. Усі хеш-функції використовують різні обсяги зсуву та додаткові константи, але їх структури практично ідентичні, відрізняються лише кількість проходів. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 23 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| SHA-256 являє собою односпрямовану функцію для створення цифрових відбитків фіксованої довжини (256 біт, 32 байт) з вхідних даних розміром до 2,31 ексабайт (2⁶⁴ біт) і є окремим випадком алгоритму з сімейства криптографічних алгоритмів SHA-2 ( Secure Hash Algorithm Version 2 ) опублікованим АНБ США (Агентством національної безпеки США) у 2002 році.  Основна робота цієї хеш-функції полягає в перетворенні (або хешування) набору елементів даних в значення фіксованої довжини. Це значення довжини буде порівнюватися з копіями вихідних даних, без можливості вилучення цих вихідних даних.  Оригінал тексту після доповнення розбивається на блоки, кожен блок - на 16 слів. Алгоритм пропускає кожен блок повідомлення через цикл з 64 ітераціями. На кожній ітерації 2 слова перетворюються, функцію перетворення задають інші слова. Результати обробки кожного блоку складаються, сума є значенням хеш-функції. Так як ініціалізація внутрішнього стану проводиться результатом обробки попереднього блоку, то немає можливості обробляти блоки паралельно.  https://cdn-images-1.medium.com/max/800/1*KvKbvq-1AIzxYRukfvMw_g.png  Рис.2.1 Графічне представлення однієї ітерації обробки блоку даних | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 25 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 2.6. Утиліта для ведення журналів Log4j  Apache Log4j - це утиліта для ведення журналів на базі Java. Вона є частиною сервісу протоколювання Apache проекту Apache Software Foundation . Log4j - це одна з декількох систем журналювання Java .  Внесення операторів коду до журналу - це метод низькорівневий метод для його налагодження. Це також може бути єдиним способом, оскільки дебагери не завжди доступні або застосовні. Це, як правило, актуально для багатопоточних додатків та розподілених програм у цілому.  Досвід показує, що ведення журналів є важливим компонентом циклу розробки. Він пропонує кілька переваг. Він забезпечує точний контекст виконання програми. Після вставки в код, генерація виходного журналу не вимагає втручання людини. Більш того, вихідний журнал можна зберегти в постійному середовищі для вивчення пізніше. На додаток до його використання в циклі розробки, достатньо багатий пакет реєстрації також може розглядатися як інструмент аудиту.  Журнал має свої недоліки. Це може уповільнити програму. Якщо це занадто багатослівне протоколювання, це може спричинити незрозумілісь протоколу програми. Щоб полегшити ці проблеми, log4j розроблений таким чином, щоб бути надійним, швидким і розширюваним. Оскільки журналювання рідко є основним завданням програми, API log4j прагне бути простим для розуміння та використання.  2.7. JSTL  JavaServer Pages Standard Tag Library ( JSTL ) є одним з компонентів Java Enterprise Edition веб - додатків платформи розробки. Це розширює | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 26 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| специфікацію JSP , додаючи бібліотеку тегів JSP для звичайних завдань, таких як обробка даних XML , умовне виконання, доступ до бази даних , цикли та інтернаціоналізація .  JSTL є альтернативою такому виду вбудованої в JSP логіки, як скріплет, тобто прямі вставки Java коду. Використання стандартизованого безлічі тегів краще, оскільки одержуваний код легше підтримувати і простіше відокремлювати бізнес-логіку від логіки відображення. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 27 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 3. АНАЛІЗ ПРОГРАМНОГО ЗАПЕСПЕЧЕННЯ  3.1. Java Virtual Machine  Віртуальна машина Java ( JVM ) являє собою віртуальну машину , яка дозволяє комп'ютеру працювати Java - програмам, а також програмам , написаних на інших мовах і інтерпретованих на Java байткод . JVM детально визначається специфікацією, яка формально описує, що вимагається від впровадження JVM. Наявність специфікації забезпечує сумісність програм Java з різними реалізаціями, так що авторам програмне потрібно турбуватися про особливі схеми базової апаратної платформи.  https://upload.wikimedia.org/wikipedia/commons/d/dd/JvmSpec7.png  Рис.3.1 Огляд віртуальної машини Java  3.2. Java cервлет  Java сервлет – це Java програма, яка розширює можливості сервера. Хоча сервлети можуть відповідати на будь-які типи запитів, вони найчастіше реалізують програми, розміщені на веб-серверах. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 28 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Сервлет Java обробляє або зберігає клас Java, який відповідає стандарту для реалізації класів Java, які відповідають на запити: API Java Servlet. Сервлети можуть в принципі спілкуватися через будь -який протокол мережі, але вони найчастіше використовуються з HTTP протоколом. Таким чином, "сервлет" часто використовується як скорочення "HTTP-сервлет". Таким чином, розробник програмного забезпечення може використовувати сервлет для додавання динамічного вмісту на веб-сервер за допомогою платформи Java . Сформований вміст зазвичай у форматі HTML, але може бути й інші дані, такі як XML, JSON. Сервлет може підтримувати стан необхідних змінних в сесійній змінних сервера з використанням HTTP cookies або URL перезапису .  Для розгортання та запуску сервлета потрібно використовувати веб-контейнер.  3.3 JavaServer Page  JavaServer Page (JSP) - це технологія, яка допомагає розробникам програм створювати динамічно створювані веб-сторінки на основі HTML , XML або інших типів документів.  JSP дозволяє Java-коду та деяким заздалегідь визначеним діям переплітатися з вмістом статичного вмісту веб-розмітки, наприклад, HTML, причому результуюча сторінка збирається та виконується на сервері. Скомпільовані сторінки, а також будь-які залежні бібліотеки Java містять Java-байт-код, а не машинний код.  Як і будь-яка інша програма Java, вона повинна бути виконана в рамках віртуальної машини Java (JVM), яка взаємодіє з операційною системою хосту сервера. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 29 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| E:\Documents\4 курс\Diploma\My\JSPLife.png  Рис.3.2 Життєвий цикл JSP  3.4. Веб-контейнер  Веб - контейнер (також відомий як контейнер сервлетів) є компонентом веб - сервера , який взаємодіє з Java сервлетами . Веб-контейнер відповідає за управління життєвим циклом сервлетів, відображення URL-адреси до певного сервлета та забезпечення того, що запитувач URL-адреси має правильні права доступу.  Основною ідеєю контейнера Servlet є використання Java для динамічного створення веб-сторінки на стороні сервера. Таким чином, контейнер сервлета по суті є частиною веб-сервера, який взаємодіє з сервлетами. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 30 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| https://www.programcreek.com/wp-content/uploads/2013/04/web-server-servlet-container.jpg  Рис.3.3 Схематичне зображення взаємодії контейнеру сервлетів з веб-сервером  Веб-контейнер обробляє запити до сервлетів, файлів JavaServer Pages (JSP) та інших типів файлів, що містять код сервера. Веб-контейнер створює екземпляри сервлету, завантажує та вивантажує сервлети, створює та управляє об'єктами запиту та відповіді, а також виконує інші завдання керування сервером.  Веб-контейнер реалізує контракт веб-компонентів архітектури Java Enterprise Edition, вказуючи середовище виконання для веб-компонентів, що включає безпеку, паралельність, керування життєвим циклом, транзакцію, розгортання та інші послуги.  Порядок обробки запиту на веб-сервері який використовує контейнер сервлетів:   1. Веб-сервер отримує HTTP-запит 2. Веб-сервер пересилає запит до контейнера сервлета 3. Сервлет динамічно завантажується та завантажується в адресний простір контейнера, якщо він не знаходиться в контейнері. 4. Контейнер викликає метод init () для ініціалізації сервлета (викликається один раз, коли сервлет завантажується в перший раз) | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 31 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1. Контейнер викликає метод service () сервлет для обробки запиту HTTP, тобто читає дані у запиті та формує відповідь. Сервлет залишається в адресному просторі контейнера і може обробляти інші HTTP-запити. 2. Веб-сервер повертає динамічно згенеровані результати до потрібного місця   https://www.programcreek.com/wp-content/uploads/2013/04/servlet-container-life-cycle.jpg  Рис.3.4 шість етапів обробки запиту на веб-сервері  Використання сервлетів дозволяє JVM обробляти кожен запит в окремому потоці Java, і це є однією з основних переваг контейнера сервлетів. Кожен сервлет - це клас Java з особливими елементами, що відповідають HTTP-запитам. Основна функція сервлету полягає в тому, щоб передати запити до сервлета для обробки і повернути динамічно згенеровані результати до правильного розташування після того, як JVM обробить їх. У більшості випадків контейнер сервлета працює з єдиною JVM, але є рішення, коли контейнер потребує декількох JVM. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 32 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 3.5. Apache Tomcat  Apache Tomcat, часто називається Tomcat Server- це контейнер Java Servlet з відкритим кодом, розроблений Apache Software Foundation (ASF). Tomcat реалізує декілька специфікацій Java Enterprise Edition, включаючи Java Servlet , JavaServer Pages (JSP), Java EL та WebSocket, і забезпечує середовище HTTP веб-сервера можливостями JVM, в якому може працювати код Java.  Tomcat був випущений разом з такими компонентами, як Catalina (контейнер для сервлета), Coyote (HTTP-коннектор) і Jasper ( JSP-двигун ). Проведемо їх аналіз та зїясуємо, які функії вони виконують.  Catalina - контейнер сервлета Tomcat . Каталіна реалізує технічні характеристики Sun Microsystems для сервлета та JavaServer Pages (JSP). У Tomcat елемент Realm являє собою "базу даних" імен користувачів, паролів і ролей (подібних до груп Unix), призначених для цих користувачів. Різні реалії Realm дозволяють інтегрувати Catalina в оточення, де така інформація про автентифікацію вже створюється та підтримується, а потім використовувати цю інформацію для реалізації керованої безпеки контейнера.  Coyote - це компонент-коннектор для Tomcat, який підтримує протокол HTTP 1.1 як веб-сервер. Це дозволяє Catalina, також виконувати роль простого веб-сервера, який обслуговує локальні файли як HTTP-документи. Coyote слухає вхідні з'єднання з сервером на конкретному порту TCP і пересилає запит до Tomcat Engine для обробки запиту та відправки назад запитуючому клієнту.  Jasper - це Tomcat JSP Engine. Jasper парсить JSP файли для компіляції їх в Java код, як сервлети (які можуть бути оброблені з допомогою Catalina). | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 33 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Під час виконання програми Jasper виявляє зміни до файлів JSP і перекомпилює їх, якщо такі були. Під час перекомпіляції модифікованого JSP-коду Java, стара версія все ще доступна для запитів сервера. Головний сервлет JSP видаляється після завершення перекомпилювання нового сервлету JSP.  Поєднання бібліотеки тегів JSP - кожна розмітка тегів у файлі JSP обробляється класом обробника тегів. Об'єкти класу обробника тегів можуть бути об'єднані та повторно використані в всьому сервлеті JSP.  Інструмент розробки Java Enterprise Edition пропонує можливісь створювати, так звані “Custom Tag”, тобто теги власноручної розробки, та їх впровадження у систему.  Кластер використовується для керування великими програмами. Він використовується для балансування навантаження, що може бути досягнуто за допомогою багатьох методів. Підтримка кластерингу вимагає JDK версії 1.5 або пізнішої версії.  Функція високої доступності використовується для полегшення планування оновлень системи (наприклад, нових випусків, запитів на зміну), не впливаючи на життя середовища. Це відбувається шляхом відправлення запитів на інший порт тимчасового сервера, тоді як головний сервер оновлюється на основному порту. Це дуже корисно для обробки запитів користувачів на веб-додатках з високим трафіком.  Веб-додатки доповняють як користувальницькі, так і системні веб-додатки, щоб додати підтримку розгортання веб-серверу в різних середовищах. Він також намагається керувати сесіями, а також програмами по всій мережі. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 34 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tomcat розробляє додаткові компоненти. Ряд додаткових компонентів може бути використаний з Apache Tomcat. Ці компоненти можуть бути побудовані користувачами, якщо вони їм потрібні, або їх можна завантажити глобального репозиторія Apache Fundation.  3.6. MySQL Server  MySQL - найпопулярніша у світі база даних з відкритим кодом. Завдяки своїй перевіреній продуктивності, надійності та простоті використання, MySQL стала провідним вибором баз даних для веб-застосунків, що використовуються високопрофесійними веб-ресурсами, включаючи Facebook, Twitter, YouTube, Yahoo! і багато інших.  MySQL - це система управління базами даних. У реляційних базах даних дані зберігаються не всі разом, а в окремих таблицях, завдяки чому можна досягти підвищення швидкості і гнучкості. Таблиці мають зв'язуються між собою за допомогою відносин, завдяки чому забезпечується можливість об'єднувати при виконанні запиту дані з декількох таблиць. SQL як частина системи MySQL можна охарактеризувати як мову структурованих запитів плюс найбільш поширений стандартний мова, яка використовується для доступу до баз даних.  MySQL є системою клієнт-сервер, яка містить багато-SQL-сервер, що забезпечує підтримку різних обчислювальних машин баз даних, а також кілька різних клієнтських програм і бібліотек, засоби адміністрування і широкий спектр програмних інтерфейсів (API). Система безпеки заснована на привілеях та паролях з можливістю верифікації з віддаленого комп'ютера, за рахунок чого забезпечується гнучкість і безпека. Паролі при передачі по мережі при з'єднанні з сервером шифруються. Клієнти можуть з'єднуватися з MySQL, використовуючи сокети TCP / IP, сокети Unix або іменовані канали. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 35 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Cтруктура MySQL трирівнева: бази даних - таблиці - записи. Логічно - таблиця являє собою сукупність записів. А записи - це сукупність полів різного типу. Ім'я бази даних MySQL унікально в межах системи, а таблиці - в межах бази даних, поля - в межах таблиці. Один сервер MySQL може підтримувати відразу кілька баз даних, доступ до яких може розмежовуватися логіном і паролем. Знаючи ці логін і пароль, можна працювати з конкретною базою даних. Наприклад, можна створити або видалити в ній таблицю, додати записи і т.д. Зазвичай ім'я-ідентифікатор і пароль призначаються хостинг провайдерами, які і забезпечують підтримку самої бази даних. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 36 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 4. АНАЛІЗ НЕОБХІДНОГО АПАРАТНОГО ЗАПЕСПЕЧЕННЯ  4.1. Характеристики веб-серверу  Плануючи серверні вимоги до обладнання для розгортання веб-серверу, буде необхідно оцінити масштабованість серверів на основі пікових відвідувачів, співвідношення максимальної кількості користувачів до пікового значення одночасних підключень та загальної навантаженості систем бізнес логіки.  Коефіцієнт перегляду – відношення кількості відвідувачів які виконують оновлення, до тих, що переглядають лише вміст сторінки.  Загальний вміст системи краще всього оцінювати за рахунком загальної пропускної спроможності.  Стійкість системи найкраще оцінюється за умови постійного потоку одночасних користувачів.  Значення, наведені нижче, стосуються мінімального наявного устаткування, необхідного для запуску Apache Tomcat наприклад, розмір мінімальної купи становить 250 МБ. Вам знадобиться додаткове фізичне обладнання, принаймні мережева карта, необхідна будь-яким додаткам, що працюють на сервері.  Мінімальна рекомендація щодо обладнання:   * процесор Intel Core 2 Duo 2.40 Ггц або вище * оперативна пам’ять обсягом 2 ГБ або більше * 64-розрядна операційна система * жорсткий диск не менше 1 ГБ | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 37 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| У особливих випадках навантаження на сервер перш за все залежить від пікових відвідувачів, тому мінімальні системні вимоги важко судити. Ці цифри надаються як керівництво до абсолютного мінімуму, необхідного для запуску сервера, і для вашої конфігурації, швидше за все, буде потрібно обладнання з кращими характеристиками.  Провівши навантажувальне тестування та проаналізувавши необхідні характеристики для запуску веб-серверу, були вирахувані приблизні показники технічних характеристик для можливих випадків одночасного виконання операцій. Зазначено, що обсяг оперативної пам'яті стосується або загальної пам'яті сервера або пам'яті, що виділяється JVM.   |  |  |  |  | | --- | --- | --- | --- | | Операції | Процесори | Процесор (ГГц) | ОЗУ (МБ) | | 150 | 1 | 2.6 | 1024 | | 250 | 2 | 2.8 | 1,536 | | 1000 | 4 | 3 | 2,048 | | 4 000 | 2 | 3.8 | 2,048 | | 6 000 | 2 | 3.6 | 4,096 | | 8 000 | 2 | 3.6 | 4,096 | | 10 000 | 4 | 2.6 | 4,096 |   Таблиця.4.1 результати навантажувального тестування  4.2. Характеристики серверу баз даних  Система пред'являє наступні вимоги до апаратного та програмного забезпечення сервера:  Мінімальні вимоги до сервера:   * 2-х ядерний процесор з тактовою частотою 2. Ггц або вище * оперативна пам’ять обсягом 4 ГБ | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 38 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| * один завантажувальний диск не менше 1 ГБ * 32-х розрядна операційна система.   Рекомендовані вимоги до сервера:   * 4-х ядерний процесор або більше * оперативна пам’ять обсягом 8 ГБ або більше * мінімум два логічних дискових елемента під’єднаних за технологією RAID10 * один завантажувальний диск не менше 1 ГБ * 64-х розрядна операційна система.   Існує також можливість інсталяції системи на платформі Windows. При цьому вимоги до попередньо програмного забезпечення залишаються незмінними.  Для роботи з клієнтською частиною системи потрібен комп'ютер, підключений по протоколу TCP / IP до мережі, в якій знаходиться.Система пред'являє наступні вимоги до апаратного та програмного забезпечення сервера.  4.2.1 Двигун InnoDB  InnoDB - це двигун загального зберігання даних, який забезпечує високу надійність та високу продуктивність. У MySQL, InnoDB є двигуном зберігання за замовчуванням. Розглянемо основні особливості:   * Його операції DML використовують модель ACID, а транзакції, що передбачаються цією моделю, дозволяють захистити дані користувачів * Рівень замикання транзакції збільшує можливості багатопоточного використання бази даних | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 39 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| * InnoDB таблиці організовують дані на диску для оптимізації запитів на основі первинних ключів (PRIMARY KEY). У кожній InnoDB таблиці є індекс первинного ключа, який називається кластерним індексом, який організовує дані для мінімізації введення / виводу та пошуку первинних ключів, використовуючи алгоритм “B-tree”. * Для збереження цілісності даних , InnoDB підтримує зовнішні ключі (FOREIGN KEY). Вставки, оновлення та видалення перевіряються за зовнішніми ключами, щоб попередити загублення даних у пов’язаних таблицях.   + 1. RAID10   RAID (надлишковий масив незалежних дисків) - конфігурація жорстких дисків, яка використовє методи розмітки , дзеркальне відображення або парності для створення великих надійних сховищ даних з декількох загального призначення комп'ютерних накопичувачів на жорстких магнітних дисках (HDD).  RAID10 - гібридний RAID, використовується для отримання додаткової продуктивності, зменшення нагрузки на носії, досягання більшої захищеності даних, в результаті об'єднання властивостей двох стандартних моделей підключення RAID.  RAID 10, як правило, реалізується RAID-контролерами, та являє собою масив дзеркальних сховищ типу RAID 0, який може являти собою двосторонні та тристоронні дзеркала та вимагає щонайменше чотирьох дисків. Тим не менш, нестандартне визначення Також можливі масиви більше чотирьох дисків. | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 40 |
| Змін. | Арк. | № докум. | Підпис | Дата |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| https://upload.wikimedia.org/wikipedia/commons/thumb/e/e6/RAID_10_01.svg/800px-RAID_10_01.svg.png  Рис.4.1 Типова конфігурація RAID10 | | | | | | |
|  |  |  |  |  | ДП.468300.004 ПЗ | Арк. |
|  |  |  |  |  | 41 |
| Змін. | Арк. | № докум. | Підпис | Дата |