МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНТЕЛЕКТУАЛЬНИХ ТЕХНОЛОГІЙ І ЗВ’ЯЗКУ

ФАКУЛЬТЕТ\_Інформаційних технологій та кібербезпеки\_

КАФЕДРА\_\_\_\_ Інженерії програмного забезпечення\_\_\_\_\_

**З В І Т**

з \_\_\_\_\_\_Виробничої\_\_\_\_\_\_\_\_ **практики**

(вид практики)

|  |  |
| --- | --- |
| Здобувача вищої освіти | Бухти Микита Миколайовича |
| Освітнього рівня | бакалавр |
| За спеціальністю | Інженерія програмного забезпечення |
| За освітньо-професійною програмою | Інженерія програмного забезпечення |
| Курс, група | 4 курс, група ІПЗ 4.04 |

Відповідальний за практику від кафедри

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(П.І.Б.) (підпис)

Керівник практики від підприємства

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(П.І.Б.) (підпис)

м. Одеса

**ЗМІСТ**

|  |  |  |
| --- | --- | --- |
| 1 | Вступ.............................................................................................................................. | 3 |
| 2 | Основи роботи з Java, Spring та Hibernate................................................................... | 4 |
| 3 | Набуті знання та навички ........................................................................................... | 7 |
| 4 | Висновки....................................................................................................................... | 8 |
| 5 | Використані джерела………………………………………………………………… | 9 |

**1.Вступ**

Виробнича практика є важливим етапом навчального процесу, який дає можливість студентам застосувати теоретичні знання в реальних умовах виробництва. Вона дозволяє поглибити знання з обраної спеціальності, здобути практичний досвід та розвинути навички роботи з сучасними технологіями. Під час практики студент працює над реальними проектами, виконує завдання, що дозволяють розв'язувати комплексні проблеми та вдосконалювати процеси в компанії. Практика є важливим етапом у розвитку професійних навичок, оскільки дає можливість студентам отримати практичний досвід роботи в команді, навички управління проектами, а також адаптації до змінних умов і вимог індустрії. Це не тільки дає можливість зрозуміти специфіку галузі, але й дозволяє набратися досвіду в плануванні, розробці та тестуванні програмних продуктів, а також вдосконалити комунікаційні та організаційні здібності.

Метою цієї практики було вивчення мови програмування Java, знайомство з основами об'єктно-орієнтованого програмування, а також освоєння фреймворку Spring для розробки сучасних веб-додатків. Практика також передбачала роботу з технологіями Інтернету речей, що дозволило студенту здобути цінний досвід в розробці програмних рішень для реальних задач. Зокрема, це включало роботу з об'єктно-реляційним відображенням за допомогою Hibernate, створенням RESTful веб-сервісів на основі Spring MVC та налаштуванням безпеки в додатках за допомогою Spring Security. Отримані під час практики знання і навички дозволяють краще зрозуміти вимоги та реалії сучасної розробки програмного забезпечення та Інтернету речей, а також підготувати студентів до роботи в умовах реального виробництва та сучасних IT-інфраструктур.

**2.** **Основи роботи з Java, Spring та Hibernate**

**1. Основи роботи з Java**

Java є однією з найбільш популярних мов програмування, яка використовується для розробки різноманітних програмних рішень — від простих додатків до складних корпоративних систем. Її основними перевагами є об'єктно-орієнтований підхід, переносимість, безпечність і стабільність. Ось основні аспекти роботи з Java:

* **Структура програми:** Програма на Java зазвичай складається з класів та об'єктів. Клас визначає шаблон для об'єкта, а об'єкт є конкретною реалізацією цього класу. Кожен клас в Java має методи та змінні для зберігання даних. Програма завжди починається з методу main, який є точкою входу в програму.
* **Типи даних та змінні:** Java підтримує різні типи даних, включаючи примітивні типи, такі як int, double, char, boolean, а також об'єктні типи, які є основою для створення класів та об'єктів. Змінні в Java використовуються для зберігання значень та працюють у контексті класів або методів.
* **Об'єктно-орієнтоване програмування (ООП):** Java є об'єктно-орієнтованою мовою, що означає використання таких принципів, як спадкування, інкапсуляція, поліморфізм та абстракція. Ці принципи дозволяють створювати гнучкі та масштабовані додатки.
* **Управління пам'яттю:** В Java використовується автоматичне управління пам'яттю за допомогою механізму збору сміття (Garbage Collection), що знижує ризик витоків пам'яті та робить розробку більш зручною.

**2. Основи роботи з Spring**

Spring є популярним фреймворком для розробки корпоративних Java-додатків. Він надає потужні інструменти для створення масштабованих і надійних систем, що зменшує складність розробки. Ось основні концепції роботи з Spring:

* **Spring Core:** Основний модуль Spring забезпечує базову функціональність, зокрема механізм інверсії керування (IoC) і впровадження залежностей (DI). Це дозволяє розробникам створювати слабо зв'язані компоненти, що спрощує тестування та обслуговування.
* **Spring Boot:** Це надбудова над Spring, яка дозволяє швидко налаштувати і запускати додатки з мінімумом конфігурацій. Spring Boot автоматично налаштовує багато параметрів, що дозволяє розробникам зосередитися на написанні бізнес-логіки, а не на налаштуванні сервера або підключеннях до бази даних.
* **Spring MVC:** Модуль для створення веб-додатків за допомогою патерну Model-View-Controller (MVC). Він дозволяє відокремлювати логіку обробки запитів, відображення даних і взаємодії з користувачем, що робить програму гнучкою і легко масштабованою.
* **Spring Security:** Надає потужний набір інструментів для захисту додатків. За допомогою Spring Security можна налаштувати автентифікацію та авторизацію користувачів, а також захистити веб-додатки від різних типів атак.
* **Spring Data:** Цей модуль спрощує роботу з базами даних, автоматизуючи багато операцій із збереженням і запитом даних. Він забезпечує інтеграцію з різними технологіями баз даних, включаючи реляційні, NoSQL бази даних, і дає можливість створювати ефективні та надійні репозиторії.

**3. Основи роботи з Hibernate**

Hibernate — це популярний фреймворк для роботи з базами даних у Java. Він дозволяє розробникам працювати з реляційними базами даних, не пишучи складні SQL-запити, завдяки використанню об'єктно-реляційного відображення (ORM). Основні моменти, які потрібно знати про Hibernate:

* **ORM (Object-Relational Mapping):** Hibernate забезпечує маппінг між об'єктами Java та таблицями бази даних. Це дозволяє працювати з об'єктами замість ручного написання SQL-запитів, що значно спрощує код і робить його більш читабельним.
* **Конфігурація Hibernate:** Для налаштування Hibernate необхідно створити конфігураційний файл, в якому вказуються параметри підключення до бази даних, а також клас-сутність (Entity), яка відображатиметься на таблицю бази даних.
* **Запити за допомогою HQL (Hibernate Query Language):** HQL — це об'єктно-орієнтована заміна SQL, що дозволяє працювати з об'єктами замість таблиць. Вона має подібний синтаксис до SQL, але замінює таблиці на класи та стовпці на властивості об'єктів.
* **Використання сесій:** Hibernate використовує концепцію сесій для взаємодії з базою даних. Сесії забезпечують доступ до операцій збереження, оновлення та видалення даних, а також для виконання запитів.
* **Кешування:** Hibernate підтримує механізм кешування, що дозволяє знижувати навантаження на базу даних і покращує продуктивність додатка, зберігаючи результати запитів у пам'яті.

**4. Інтеграція Java, Spring та Hibernate**

Коли ці три технології працюють разом, розробники можуть створювати високопродуктивні та масштабовані корпоративні додатки. Spring надає основу для структурування додатків, зокрема через впровадження залежностей та управління життєвим циклом об'єктів, в той час як Hibernate забезпечує ефективну роботу з базами даних. Spring Boot спрощує налаштування середовища, а Spring Data дозволяє інтегрувати Hibernate без необхідності вручну писати SQL-запити.

**3. Набуті знання**

Під час практики було здобуто ряд важливих теоретичних та практичних знань, які стосуються основ програмування на Java, роботи з фреймворком Spring та використання Hibernate для роботи з базами даних. Ось основні моменти, які були освоєні:

1. **Основи програмування на Java:** Вивчено основи синтаксису мови Java, структуру програми, базові типи даних, оператори, умовні конструкції, а також методи та їх застосування. Приділено увагу об'єктно-орієнтованим принципам, таким як інкапсуляція, спадкування, поліморфізм і абстракція, що є основою для розробки якісного та масштабованого коду.
2. **Робота з фреймворком Spring:** Ознайомлено з базовими принципами роботи з фреймворком Spring, зокрема механізмом впровадження залежностей (Dependency Injection) та інверсії керування (IoC). Вивчено роботу з Spring Boot для швидкого налаштування та запуску додатків, а також із Spring MVC для створення веб-додатків. Опановано налаштування безпеки в додатках за допомогою Spring Security.
3. **Робота з Hibernate:** Здобуто знання про об'єктно-реляційне відображення (ORM) за допомогою Hibernate. Вивчено налаштування Hibernate для роботи з базами даних, створення сутностей та використання HQL для виконання запитів. Опановано основи кешування та оптимізації роботи з базою даних.
4. **Інтеграція технологій:** Була набуті практичні навички інтеграції технологій Java, Spring та Hibernate для створення веб-додатків, що включають роботу з базами даних. Застосовано вивчені технології для розв'язання реальних завдань, таких як створення RESTful сервісів, обробка даних з бази даних через репозиторії, налаштування безпеки та авторизації.
5. **Робота в команді та управління проектами:** Під час практики здобуто досвід роботи в команді, участі в розробці програмних рішень та покращенні внутрішніх процесів. Залучено до організації комунікації в команді, що дозволило покращити продуктивність та ефективність роботи.

Завдяки цим знанням, отриманим під час практики, вдалося успішно реалізувати поставлені завдання та отримати цінний досвід для подальшої роботи в розробці програмного забезпечення.

**Висновки**

Протягом практики здобувач вищої освіти отримав цінний досвід у розробці програмних рішень на мові Java, а також в освоєнні сучасних фреймворків, таких як Spring та Hibernate. Це дозволило здобути знання, які є основою для розробки масштабованих, надійних та ефективних додатків.

Під час практики студент здобув досвід роботи з основними концепціями об'єктно-орієнтованого програмування, такими як інкапсуляція, спадкування та поліморфізм, а також опанував роботу з інструментами для створення веб-додатків. Завдяки роботі з Spring та Hibernate студент навчився інтегрувати різні компоненти системи, працювати з базами даних та забезпечувати безпеку додатків.

Додатково, практикант проявив себе як відповідальний та ініціативний працівник, активно долучався до командної роботи та покращення внутрішніх процесів. Його здатність швидко освоювати нові технології та застосовувати їх на практиці свідчить про високий рівень професіоналізму та готовність працювати в реальних умовах розробки програмного забезпечення.

Узагалі, під час проходження практики здобувач продемонстрував високий рівень знань, умінь та готовності до подальшої професійної діяльності в галузі розробки програмного забезпечення. Це дозволяє з упевненістю говорити про його успішне завершення практики та готовність до виконання більш складних завдань у майбутньому.

**Використані джерела**

1. Офіційна документація Java:

* Oracle. (n.d.). The Java™ Tutorials. https://docs.oracle.com/javase/tutorial/

1. Spring Framework:

* Spring Framework Documentation. (n.d.). Spring Framework. https://spring.io/docs
* Spring Boot Documentation. (n.d.). Spring Boot. https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/

1. Hibernate ORM:

* Hibernate ORM Documentation. (n.d.). Hibernate. https://hibernate.org/orm/documentation/

1. Книги:

* Гаврилюк, В. А. (2020). Java. Базовий курс програмування. Київ: Літера.
* Фаулер, М. (2018). Patterns of Enterprise Application Architecture. Addison-Wesley.

1. Онлайн ресурси:

* Stack Overflow. (n.d.). Java, Spring, Hibernate discussions and Q&A. https://stackoverflow.com
* Baeldung. (n.d.). Spring Tutorials. https://www.baeldung.com

