Міністерство освіти і науки України
Державний університет «Одеська політехніка»
Інститут інформаційної безпеки, радіоелектроніки та телекомунікацій
Кафедра кібербезпеки та програмного забезпечення

Перерва Станіслав Олегович,
здобувач групи РФ-161

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

Розробка захищеної автоматизованої веб-системи тестування знань

Спеціальність:
122 Комп’ютерні науки

Спеціалізація, освітня програма:
Програмне забезпечення систем захисту інформації

Керівник:
Мокріцький Вадим Анатолійович,
д.т.н., проф.

Одеса – 2021

АНОТАЦІЯ

Кваліфікаційна робота на тему «Розробка захищеної автоматизованої веб-системи тестування знань» на здобуття другого (магістерського) рівня вищої освіти за спеціальністю 122 – Комп’ютерні науки, спеціалізація, освітня програма: Програмне забезпечення систем захисту інформації, містить: 4 рисунка, 1 таблицю, 1 додаток, 52 літературних джерела за переліком посилань. Робота виконана на 71 сторінках загального тексту і 48 сторінках основного тексту.

Метою даної роботи є підвищення рівня інформаційної безпеки розробленої та реалізованої автоматизованої веб-системи тестування знань.

При розв’язанні поставлених в кваліфікаційній роботі теоретичних задач були використані наступні методи досліджень: аналіз педагогічної літератури з проблеми системного контролю якості знань; технології реалізації системного контролю; аналіз найбільш критичних ризиків та існуючих механізмів захисту веб-додатків. Програмна частина роботи виконана із застосуванням сучасних технологій побудови веб-додатків.

Результатом кваліфікаційної роботи є розробка та реалізація автоматизованої веб-системи тестування знань, рівень інформаційної безпеки якої, забезпечує захист інформації від найбільш критичних ризиків, встановлених у відповідності до OWASP Топ-10 2021. Розроблений програмний продукт може бути рекомендований для застосування в вищих та загальноосвітніх навчальних закладах, для підвищення якості знань та об’єктивності, завдяки застосуванню форми проведення тестування у автоматизованому форматі, з підвищеним рівнем інформаційної безпеки.

ЗАХИСТ ІНФОРМАЦІЇ, ВЕБ-ЗАСТОСУВАННЯ, ДИСТАНЦІЙНЕ НАВЧАННЯ, ЯКІСТЬ ОСВІТИ, ТЕСТУВАННЯ ЗНАНЬ

ABSTRACT

Qualification work «Development of a secure automated web-based knowledge testing system» for the second level of higher education (master's) in the specialty 122 – Computer Science, specialization, educational program: Software for information security systems, contains: 4 figures, 1 table, 1 appendix, 52 references according to the list of references. Work carried out on 71 total pages of text and 48 pages of main text.

The aim of this work is to increase the level of information security of the developed and implemented automated web-based knowledge testing system.

For the solving theoretical problems posed in the qualification work, the following research methods were used: analysis of pedagogical literature on the problem of systematic quality control of knowledge; technologies for the implementation of system control; analysis of the most critical risks and existing web application protection mechanisms. The software part of the work was done using modern technologies for building web applications.

Result of the qualification work is the development and implementation of an automated web-based knowledge testing system, the level of information security of which ensures the protection of information from the most critical risks established in accordance with the OWASP Top 10 2021. The developed software product can be recommended for use in higher and general educational improving the quality of knowledge and objectivity, thanks to the application of the form of testing in an automated format, with an increased level of information security.

INFORMATION PROTECTION, WEB APPLICATIONS, DISTANCE LEARNING, THE QUALITY OF EDUCATION, KNOWLEDGE TESTING

ЗМІСТ

ВСТУП

Сучасна освіта націлена на надання можливості отримання якісних знань та умінь. Багато чисельні дослідження останніх років демонструють значний вклад інформаційних технологій в підвищення якості отримуваних знань та умінь в будь-яких сферах освіти. Для оцінки якості отримуваних знань на протязі учбового процесу необхідно виконувати проведення контролю у різних формах. Значні педагогічні дослідження доводять ефективність проведення тестового контролю для отримання оцінки якості знань та умінь. Інформаційні технології можуть допомогти підвищити незалежність перевірки та встановлення оцінки від особи викладача, тобто підвищити об’єктивність, завдяки застосуванню форми проведення тестування у автоматизованому форматі.

У зв’язку з цим, розробка захищеної автоматизованої веб-системи тестування знань є надзвичайно актуальною. Компанії з розробки програмного забезпечення найчастіше не дотримуються вимог інформаційної безпеки, концентруючись на інших цілях. Компанія Positive Technologies постійно виконує дослідження з інформаційної безпеки веб-додатків, які свідчать, що значна частина програмного веб-забезпечення містить вразливості різного ступеня ризиків. OWASP – відкритий проект з дослідження безпеки веб-додатків, щорічно публікує топ десять найпоширеніших загроз.

Тому, метою даної роботи є підвищення рівня інформаційної безпеки розробленої та реалізованої автоматизованої веб-системи тестування знань.

Для досягнення поставленої мети необхідно розв’язати наступні задачі:

а) виконати аналіз педагогічного напрямку дослідження форми проведення та оцінки якості знань у форматі автоматизованого тестування;
б) виконати аналіз найбільш критичних ризиків та існуючих механізмів захисту веб-додатків;
в) розробити механізм захисту програмного забезпечення, рівень інформаційної безпеки якої, забезпечує захист інформації від найбільш критичних ризиків, встановлених у відповідності до OWASP Топ-10 2021;

г) реалізувати програмний продукт, автоматизовану веб-систему тестування знань, з підвищеним рівнем інформаційної безпеки, у відповідності до запропонованого механізму захисту програмного забезпечення.

Об'єктом дослідження є програмне забезпечення, що зберігається, оброблюється й передається в комп'ютерних системах і мережах.

Предметом дослідження є програмні методи захисту програмного забезпечення від несанкціонованих змін.

При розв’язанні поставлених в кваліфікаційній роботі теоретичних задач були використані наступні методи досліджень: аналіз педагогічної літератури з проблеми системного контролю якості знань; технології реалізації системного контролю; аналіз найбільш критичних ризиків та існуючих механізмів захисту веб-додатків. Програмна частина роботи виконана із застосуванням сучасних технологій побудови веб-додатків.

Наукова новизна кваліфікаційної роботи полягає в наступному: розроблено механізм захисту програмного забезпечення, рівень інформаційної безпеки якої, забезпечує захист інформації від найбільш критичних ризиків, встановлених у відповідності до OWASP Топ-10 2021.

Результатом кваліфікаційної роботи є розробка та реалізація автоматизованої веб-системи тестування знань, рівень інформаційної безпеки якої, забезпечує захист інформації від найбільш критичних ризиків, встановлених у відповідності до OWASP Топ-10 2021. Розроблений програмний продукт може бути рекомендований для застосування в вищих та загальноосвітніх навчальних закладах, для підвищення якості знань та об’єктивності, завдяки застосуванню форми проведення тестування у автоматизованому форматі, з підвищеним рівнем інформаційної безпеки.

Робота складається із вступу, чотирьох розділів, висновку, переліку посилань та додатку. У вступі обґрунтовано актуальність дослідження, сформульована мета роботи, перераховані завдання для досягнення поставленої мети. У першому розділі виконано огляд педагогічного напрямку дослідження форми проведення та оцінки якості знань у форматі автоматизованого тестування.

Представлено особливості проведення тестувань знань, де перелічено проблеми проводження тестування в університеті та поставлено цілі на розробку програмного додатку, що буде мати переваги, бути безпечним, ефективним та впровадженим в Інтернет. Розглянуто дистанційне навчання та застосування веб-технологій. Представлені аспекти та переваги електронного оцінювання. Досліджені альтернативні системи, що мають як переваги, так і недоліки.

У другому розділі представлені теоретичні основи захисту веб-додатків. Безпеці веб-додатків необхідно приділяти дуже багато уваги. Безпека веб-додатків залежить від якості програмного коду, від кваліфікації системного адміністратора та від компетенцій усіх користувачів, які мають доступ до чутливої інформації. Вразливості OWASP є дуже шкідливими і в цьому списку зосереджені найнебезпечніші вразливості, які можуть коштувати купу грошей, або підрив ділової репутації, чи довести аж до втрати бізнесу. Методи самозахисту веб-додатку під час виконання мають деякі складнощі з виявленням вразливостей без штучного налаштування, але з використанням машинного навчання можуть бути корисними у нагоді.

У третьому розділі представлена практична реалізація захищеної автоматизованої веб-системи тестувань знань. Детально описано вибір засобів реалізації програмного продукту: особливості мов програмування JavaScript та PHP, фреймворків Nuxt та Laravel, інтегрованого середовища розробки PHPStorm. Розглянута архітектура програмного продукту «SaveTestingKnowledge». Представлені дві технології, що допомогли в вирішенні проблем, які виникли при розробці програмного продукту «SaveTestingKnowledge» при взаємодії між клієнтською та серверною частинами. Наведені основні складові програмного продукту та інструкція з експлуатації.

В четвертому розділі проаналізовані умови праці і вибір заходів і засобів захисту від небезпечних і шкідливих виробничих факторів. Виконано детальне дослідження біоритмів спеціаліста з інформаційної безпеки.

Основні положення та результати поданої роботи представлені в збірнику наукових праць «Актуальні наукові дослідження в сучасному світі».

1 ТЕСТУВАННЯ ЗНАНЬ

1.1 Особливості проведення тестувань знань

Зараз майже всі університети нашої країни використовують ручну процедуру проведення тестування знань для студентів. Ручна процедура означає, що з кожним тестуванням студенти повинні відвідувати університет, щоб взяти участь у ньому в певний час. Після цього лектори збирають листи пройденого тестування. Іноді студенти не відвідують університет з певної причини, і викладачі припускали, що студенти відсутні на цьому тестуванні. Цей сценарій несправедливий для студента. Таким чином, відповідним рішенням цієї проблеми є розробка системи тестування знань студентів з будь-якого місця.

Проблеми проводження тестування в університеті:

- викладачам потрібно більше часу приділяти оцінці. Це тому, що викладач повинен перевіряти студентські роботи по черзі;
- учням важко відповісти на тестування з будь-якого місця. Тестування впливає результуючу оцінку;
- іноді у деяких студентів є своя причина, чому не можуть пройти тест вчасно.

Поставлені цілі:

- розробити систему, яка дає змогу учням відповідати на тестування онлайн;
- запровадити систему, яка дає змогу студенту відповідати на тестування з будь-якого місця;
- перевірити систему, чи зможе вона допомогти викладачам у проведенні тестування та економити час.

Очікується, що ця система буде впроваджена в мережі Інтернеті. Крім того, ця система більш безпечна, оскільки буде відповідати вимогам захисту від найбільш потенційних вразливостей веб-додатків. Крім того, ця система також менш трудомістка і є більш ефективною. Це тому, що аналіз буде дуже простим у запропонованій системі, оскільки вона автоматизована. Результат буде дуже

точним і буде оголошено за короткий проміжок часу, оскільки розрахунок і оцінки виконується системою автоматично.

1.2 Дистанційне навчання та застосування веб-технологій

Електронне оцінювання є важливою частиною будь-якої платформи електронного навчання, оскільки вони перевіряють досягнення учнями цілей у навчанні.

Електронне оцінювання визначається як використання веб-технологій для цілей формуючого та підсумкового оцінювання в контексті формального навчання на університетському рівні. Цей термін відноситься до всього циклу процесу оцінювання, від розробки завдань оцінювання до зберігання та управління продуктами оцінки. Нормативне електронне оцінювання стосується використання технології для підтримки ітераційного процесу аналізу інформації про навчання студентів та їх оцінювання щодо попередніх досягнень результатів навчання, тоді як підсумкове електронне оцінювання пропонує докази досягнень учнів, що вони знають, розуміють і вміють робити, приписуючи цінність своїм очевидним досягненням. Вільям і Блек зауважують, що «всі оцінки можуть бути підсумковими... але лише деякі мають додаткову здатність виконувати формуючі функції», перш ніж продовжувати наголошувати на важливості концепцій валідності та надійності оцінки [1].

Електронне оцінювання є важливою частиною будь-якої платформи електронного навчання, оскільки вони перевіряють досягнення учня досягнення цілей навчання. Вони корисні в двох напрямках, як для студента, так і для викладача. Один з них призначений для зміцнення своїх знань, включаючи в першу чергу мету учня під час вивчення курсу, а інший – оцінку розуміння курсу учнями (залучаючи як учня, так і викладача, як засіб самооцінки та відстеження прогресу в порівнянні з оцінювання прогресу та дефектність навчальних матеріалів). Тому розробка методів оцінки знань є проблемою, якій постійно приділяється увага в аналізі освітніх даних.

Зростає значення та застосування інформаційно-комунікаційних технологій (ІКТ) у навчанні. Освітні технології постійно змінюються, щоб йти в ногу з ширшим соціальним розвитком і задовольняти потреби студентів, більшість з яких є цифровими вихідцями. Ця рівновага залежить від зростаючого ринку інструментів, які поєднують освіту з наявністю технологій: відкриті та зручні платформи, такі як програми для читання електронних книг і масові відкриті онлайн-курси. Втілення нової доби навчання, ці канали спрямовані на подолання географічних бар'єрів та покращення якості життя за допомогою доступних методів навчання. Дослідження показали, як ці канали покращують академічну успішність, а деякі навіть припускають позитивну кореляцію між використанням комп'ютера та успішністю в школі або університеті. Що стосується викладання, цифрове навчання надає безліч нових інструментів для відстеження прогресу в навчанні учня. Наприклад, онлайн-вікторини, або скорочено електронні вікторини, допомагають вчителям оцінювати успішність учнів, надаючи їм миттєвий зворотний зв'язок, заощаджуючи економічні витрати на друк тестових копій, виконання тестів та оцінювання їх окремо [2].

Інформаційні та комунікаційні технології протягом тривалого часу були інструментом допомоги в освіті. Використання технології в оцінці почалося в 1920-х роках, коли Сідні Л. Пресс розробив машини для автоматичного тестування. Більше того, водночас у школах почали використовувати стандартизоване оцінювання та технологію автоматичного підрахунку балів, що допомогло зробити масштабне тестування зручним та економічно ефективним. Величезні зміни в багатьох секторах, особливо в освіті, відбулися, коли в 1990-х роках було введено всесвітню мережу. З того часу багато компаній запровадили власну систему електронного оцінювання. В Англії, Уельсі та Північній Ірландії принципи та вказівки щодо електронного оцінювання були запроваджені JISC (Об'єднаний комітет інформаційної системи) для роз'яснення різних кваліфікаційних регуляторів у Сполученому Королівстві. У 2009 році IMS Global Learning Consortium розробив специфікацію інтероперабельності IMS Question

and Test. У 2009 році Cisco, Intel та Microsoft випустили Transforming Education: Assessing and Teaching 21st Century Skills [3].

Після введення електронного навчання та електронного оцінювання процес навчання розвивався. Електронне оцінювання покращило вимірювання результатів учнів і дозволило їм отримати негайний і прямий зворотний зв’язок. Важливо створити систему оцінювання студентів, яка б враховувала освітні цілі та допомагала б учням розвивати свої навички, які будуть корисними для суспільства в довгостроковій перспективі.

Електронне оцінювання може мати різні форми, такі як автоматичні адміністративні процедури, оцифрування паперових систем та онлайн-тестування, яке включає тести з множинним вибором та оцінку навичок розв’язування проблем. Сіттісак та ін. вказали, що електронне оцінювання включає підтримку оцінювання за допомогою комп'ютера, наприклад: за допомогою веб-інструментів оцінювання. Проте Реджу та Адезіна пояснюють, що електронне оцінювання включає наскрізні електронні процедури оцінювання. Це підтверджує PingSoft у своєму поясненні, що дизайн системи включає повний процес перевірки, що включає пропозицію, складання документів, підписання, перевірку, групування, статистику та аналіз. Крім того, JISC [3] визначив електронне оцінювання як наскрізний електронний процес оцінювання, при якому інформаційно-комунікаційні технології використовуються для всіх процесів оцінювання від подання запитань до збереження відповідей учнів. Більшість досліджень сходяться на думці, що електронне оцінювання – це електронне оцінювання, в якому всі процедури оцінювання від початку до кінця оцінювання мають проводитися в електронному вигляді. Це означає, що проектування, впровадження тесту, запис відповіді та надання зворотного зв’язку завершуються за допомогою ІКТ [3].

1.3 Аспекти та переваги електронного оцінювання

У нашому цифровому світі технології широко поширені в освіті. Вони підтримують виконання всіх етапів і заходів у процесі навчання. Спочатку ідея полягала в технологіях прискорення та автоматизації навчальної діяльності, але тепер перед ними стоїть набагато важливіше завдання – створити передумови для реалізації нових педагогічних підходів. Важливим етапом навчально-виховного процесу є оцінка учнями набутих знань, умінь і компетенцій. Оцінювання не тільки кількісно оцінює досягнення учнів, але й допомагає покращити навчальний процес загалом. Студенти можуть своєчасно реагувати і змінювати спосіб навчання, щоб покращити результати навчання та бути більш відданими й відповідальними до свого навчання. Оцінювання допомагає перевірити ефективність стратегій навчання, які використовуються тренерами, та сприяє їх покращенню. Оцінювання допомагає встановити, чи досягнуто і в якій мірі поставлені навчальні цілі, які прогалини в знаннях учнів. На основі цієї інформації вчителі можуть вживати заходів та дій, щоб своєчасно запобігти упущенням. Сучасні інформаційно-комунікаційні технології (ІКТ) дозволяють навчатися в цифровому середовищі з потенціалом для диверсифікації, збагачення та розвитку традиційних методів оцінювання. Часто в традиційному навчанні (без застосування ІКТ) кращим варіантом оцінювання є використання електронних вікторин. Вони дозволяють багаторазове використання запитань, швидке та легке оновлення, модифікацію та збагачення. Електронні вікторини пропонують миттєвий зворотній зв’язок, автоматичне, швидке й об’єктивне оцінювання з попередньо встановленими критеріями за допомогою шаблонів. Доступні програмні рішення підтримують багаторазові спроби учнів в оцінювальній діяльності досягти бажаного результату. Проблема полягає в тому, які інструменти використовувати для оцінювання в цифровому середовищі, щоб забезпечити прихильність і високий рівень мотивації учнів. Чи можливо і як їх використовувати для оцінки різних аспектів і рівнів знань і навичок учнів? Метою поточної роботи є представлення деяких інноваційних інструментів оцінювання в системах управління навчанням (LMS), проілюстрованих LMS Moodle. Вони дозволяють реалізувати нові підходи до оцінки, які відрізняються від

традиційних. Вони доводять тезу про те, що електронне оцінювання не еквівалентне традиційному оцінюванню, яке проводиться в цифровому середовищі.

Електронне оцінювання надає нові привабливі можливості та методики оцінювання різних аспектів і рівнів набутих учнями знань, умінь і компетенцій. Це допомагає подолати деякі недоліки, які притаманні традиційним підходам до оцінювання [4].

Електронне оцінювання передбачає використання комп’ютерів у будь-якій діяльності, пов’язаній з оцінюванням знань учнів [4].

Електронне оцінювання як процес, що включає планування, підготовку, впровадження, запис та подальший аналіз, які виконуються в електронному вигляді.

Електронне оцінювання має ряд переваг [4]:

- використання різних методик оцінювання, які відповідають заздалегідь поставленим цілям навчання та використовуються педагогічні підходи;
- оцінка великої кількості учнів швидко і легко завдяки високому ступеню процесу автоматизації;
- негайний персоналізований зворотній зв'язок, який може направляти учнів;
- об'єктивність за рахунок автоматизації оціночної діяльності або оцінювання на основі заздалегідь встановлених критеріїв і шаблонів;
- впровадження в будь-який час і в будь-якому місці, що гарантує більшу гнучкість;
- можливість повторного використання діяльності або їх компонентів у різних комбінаціях і контекстах;
- широкий вибір типів запитань в електронних вікторинах;
- можливість оцінити групу учнів у цілому, враховуючи індивідуальний внесок.

У цифровому середовищі можна проводити різні види електронного оцінювання [4]:

- за поставленими навчальними цілями та періодом, в який воно проводиться, оцінювання може бути: діагностичним (допомагає поставити чіткі, точні та адекватні навчальні цілі, що відповідають рівню готовності учнів до роботи з новим змістом); формуюча (допомагає контролювати успішність учнів та вносити зміни в навчальний процес, щоб адаптуватися до їхніх потреб та покращити навчання); підсумковий (підтримує оцінку знань і навичок учнів наприкінці навчання за встановленими критеріями або стандартами, щоб визначити, чи були досягнуті поставлені цілі навчання [5]);
- відбувається оцінювання одночасно чи ні: синхронно (одночасно оцінюються всі учні, і в більшості випадків оцінювання проводить викладач); асинхронно (кожний учень включається до оцінювання в інший, зручний для нього час);
- за кількістю оцінюваних учнів та суб'єктів-оцінювачів: групові (оцінюються вироби, які є результатом спільної роботи групи учнів та їх уміння працювати в команді); індивідуальний (оцінюються індивідуальні знання та вміння кожного учня); рівноправний (допомагає розвивати критичне мислення, мотивувати та відстоювати власну думку на основі фактів і доказів).

Важливо підтримувати учнів у повторному навчанні у відповідний час, щоб допомогти їм запам'ятати те, чого вони навчилися, надовго та ефективно [6].

1.4 Аналітика дистанційного навчання

Аналітика навчання – це нова область досліджень, що розвивається, яка має справу з даними, згенерованими під час взаємодії студентів із навчальним середовищем онлайн. Зібрані та проаналізовані фрагменти інформації щодо їхньої поведінки, продуктивності та уподобань можуть бути використані для покращення та оптимізації викладання, навчання та оцінювання.

Оцінювання завдань електронного оцінювання все ще залишається складним питанням, яке вказує на широкий спектр дослідницьких проблем, подібних до: зв’язку між змістом, структурою та поданням об’єктів навчання та оцінювання, впливу стилів навчання на об’єкти оцінювання, типу наданого зворотній зв'язок та його відповідність отриманим знанням. Перетинаючи наукові репозитарії, було знайдено лише декілька наукових праць на подібну тематику, що свідчить про необхідність подальшого дослідження та опрацювання.

Процес оцінювання як дуже важлива частина навчальної діяльності має бути розроблений правильно і точно. Вікторини потрібно розробляти, доповнювати та вдосконалювати, щоб вони були зрозумілішими, щоб дати учням можливість максимально повно показати свої знання. При розробці онлайн-вікторин необхідно враховувати той факт, що під час їх виконання учні не можуть задати конкретні запитання, а також отримати на них відповіді.

Онлайн-вікторини є викликом не лише для студентів – як добре виступити, а й для вчителів – як і в якій формі створити оцінювальні заходи, які дозволять учням показати свої знання без зайвих перешкод і труднощів.

Деякі студенти працюють і навчаються одночасно. Зворотній зв’язок є важливим для кращого сприйняття та розуміння питань вікторини та, відповідно для кращої успішності учнів у цьому типі оцінювання. Методологія дослідження складається з наступних процедур [7]:

- дослідження наукової літератури, яке здійснюється за допомогою пошукових запитів у наукових пошукових системах Google Scholar і Google Semantics та наукових базах даних Scopus і Web of Science;
- проведення онлайн-тестування студентів з двох кафедр ЦСЄ у двох різних системах електронного навчання: Moodle з відкритим кодом та хмарному Edu20. Оцінювані вікторини бувають двох типів: екзаменаційні вікторини, які сприяють отриманню підсумкової оцінки учня, та тести для самоперевірки, які використовуються для покращення успішності навчання;

- розробка інструменту опитування з метою зібрати оцінку студентів щодо онлайн-вікторин та врахування широкого спектру питань, розподілених на кілька груп. Метою є отримання інформації як для студентів, так і для їх оцінки/уявлень про проведені оцінювальні заходи;
- обговорення та аналіз отриманих результатів.

Розробка моделі оцінювання якості завдань онлайн-оцінювання у формах вікторин. Відповіді студентів дають нам можливість створити об'єктивну основу для оцінювання електронного оцінювання, яка складається з кількох рівнів:

- перший шар включає зміст, стиль формату та тип питань і відповідей;
- другий рівень відображає адекватні знання учнів щодо змісту запитань;
- третій рівень описує роль зворотного зв'язку для навчання учнів;
- четвертий рівень стосується зв'язку між стилями навчання та типом і форматом запитання.

1.5 Альтернативні працюючі системи

Reading Battle виросла з колишнього дослідницького проекту, який фінансується Гонконгським фондом якісної освіти, під назвою «Зміцнення здатності учнів до розуміння читання шляхом розробки банку електронних вікторини для дітей у хмарі» (Чу, 2016). У базі даних equiz понад 5000 питань розроблено для приблизно 500 дитячих книг. Колекція дитячої літератури в базі даних охоплює широкий спектр жанрів, охоплюючи художню літературу, казки, фольклор, науку та історію. Студенти-користувачі можуть отримати бали, кидаючись у «битви», що складаються з десяти запитань на розуміння прочитаного в кожному. Таблиця лідерів системи показує рейтинги користувачів на основі їхньої продуктивності (наприклад, загальний бал та покращення), а відмінні гравці будуть винагороджені значками та жетонами. Окрім рейтингу, значків та нагород, панель керування, дикторський текст та анімація також роблять Reading Battle цікавою та привабливою платформою для читання [2].

Відмінною особливістю LMS (learning management system) є можливість відстежувати та аналізувати результати оцінювання та прогрес учнів у навчанні. Це значно відрізняє їх від інших програмних додатків для оцінки. За допомогою наявних інструментів викладачі отримують інформацію та графічну візуалізацію досягнутих результатів і можуть зробити поглиблений аналіз. LMS охоплюють весь процес навчання. Вони збирають, зберігають і надають усі необхідні дані для успішності та діяльності учнів, які можна використовувати для підвищення якості навчання.

Інноваційні інструменти для електронного оцінювання в системах управління навчанням. LMS Moodle є одним з найпопулярніших і використовуваних LMS з відкритим кодом, який підтримується спільнотою користувачів і розробників. Його функціональність можна розширити за допомогою додаткових плагінів, у тому числі для електронного оцінювання. Перевагою системи є можливість використовувати навчальні ресурси та діяльність для створення запитань вікторини або проведення оцінювальних заходів. Значна частина цих заходів виконується автоматично, без участі викладачів. Гейміфікація інтегрує ігрові підходи, елементи та мислення в діяльність, яка не є іграми (у даному випадку в навчальній діяльності), для вирішення проблем і сприяння навчанню. LMS Moodle пропонує можливості для гейміфікації та освітніх ігор, як [7]:

- навчальна діяльність – учні отримують бали за свої дії та відкривають додаткові ресурси, коли досягають певного рівня;
- блоки – відображається поточний рівень учнів та їх прогрес до наступного;
- оцінювання діяльності – учні грають в ігри, демонструють свої знання та отримують оцінки.

В даному розділі виконано огляд педагогічного напрямку дослідження форми проведення та оцінки якості знань у форматі автоматизованого тестування.

Представлено особливості проведення тестувань знань, де перелічено проблеми проводження тестування знань в університеті та поставлено цілі на

розробку програмного додатку, що буде мати переваги, бути безпечним, ефективним та впровадженим в мережі Інтернет.

Розглянуто дистанційне навчання та застосування веб-технологій, що дозволяє студентам навчатися дома, оцінювати освоєний новий матеріал нестандартним способом, але головне – цікавим.

Представлені аспекти та переваги електронного оцінювання дозволяють економити час як викладачів, так і студентів; економити витрати на канцелярські товари; впроваджувати нові стратегії навчання та покращувати навчальний процес.

Вивчена аналітика дистанційного навчання наголошує, що онлайн-навчання є викликом як для студентів, так і для викладачів, тому що необхідно розуміти в якій формі створювати оцінювальні заходи, щоб вони були максимально ефективними та зрозумілими.

Досліджені альтернативні працюючі системи мають як свої переваги, так і недоліки. Впроваджена система буде корисною в першу чергу тим, що всі дані, що зберігаються в базі даних залишаються конференційними, а доступ до них мають лише уповноважені особи.

2 ТЕОРЕТИЧНІ ОСНОВИ ЗАХИСТУ ВЕБ-ДОДАТКІВ

2.1 Безпека веб-додатків

На сьогодні інформаційні технології стають все більш поширеними в суспільстві. Вони використовуються в найрізноманітніших галузях: у соціальних мережах, електронній пошті, новинних та урядових порталах, електронній комерції, форумах, блогах та інших веб-сайтах. Разом з розвитком ІТ все більшого значення набуває інформаційна безпека (ІБ). Якщо розглядати історичну перспективу розвитку інформаційних технологій, то виявляється, що з 1996 року, коли була прийнята Конституція нашої держави, і до моменту написання цієї роботи кількість людей, які відчувають себе захищеними в інформаційному полі, постійно падає. Справа не лише в лавиноподібному зростанні використання ІТ, а, швидше за все, у порушеннях, які держава робить в інформаційній сфері. Зростання вимог до забезпечення інформаційних технологій визначається, насамперед, розвитком ІТ. Потенційні порушники мають нові можливості для деструктивних дій у всіх сферах нашого життя. Найефективніші з цих дій відбуваються у кіберпросторі. Виходячи з вищесказаного, можна зробити висновок про необхідність захисту веб-додатків.

Залежно від того, де використовується веб-додаток, в ньому може оброблятися інформація різних рівнів доступу та значень. Наприклад, програми можуть використовувати інформацію про банківські картки, персональні дані користувачів, паролі та інші ідентифікаційні дані. Жоден користувач не застрахований від того, що його дані можуть бути скопійовані або піддані ряду випадкових і зловмисних впливів у процесі їх обробки, передачі та зберігання [8].

Веб-додатки реалізують архітектуру «клієнт-сервер». У цій концепції клієнт – це браузер користувача, а сервер – веб-сервер. В основі концепції «клієнт-сервер» відбувається обмін інформацією через мережу, клієнт починає взаємодіяти, а дані зберігаються переважно на сервері.

В основному веб-додатки мають розподілену структуру. Основні переваги цієї конструкції:

- хороша масштабованість – можливість збільшення функціональності без зміни структури;
- наявність можливості керувати навантаженням програми – направлення потоків запитів користувача на менш завантажені сервери.

Типова архітектура цих програм може бути представлена у трьох рівнях: клієнтська частина (веб-браузер), веб-сервер та база даних.

Кожен день будь-який веб-сайт може піддаватися кібератакам. Як правило, більшість атак є цільовими. Це означає, що зловмисник не обмежується однією спробою отримати необхідні дані несанкціонованим способом, оскільки він не знає, які саме уразливості є в коді. Усі спроби зламати сайт зводяться до серії подій, які відбуваються протягом певного періоду часу. За статистикою за 2018 рік, найбільша кількість атак на одне веб-додаток припадає на сайти фінансових організацій, транспортних компаній та обслуговуючих компаній. Вибір методу атаки залежить від особливостей веб-додатка. Припустимо, якщо в додатку не передбачена можливість введення даних користувача, то зловмисник не буде проводити атаки, спрямовані на зміну логіки програми за допомогою введення будь-яких даних користувачем. Найпопулярніші атаки включають: введення коду SQL, вихід за межі каталогу та між-сайтові сценарії.

Щоб виявити вразливості в коді та зрозуміти, як система реагуватиме на атаку – програму потрібно протестувати. Процес тестування максимально схожий на процес злому, який проводить зловмисник. Мета таких дій – визначити, наскільки вразливим є веб-додаток.

Для критичної інфраструктури спеціалізовані фішери використовують передові методи, які поєднують соціальну інженерію, орієнтуючись як на відсутність спеціалізованих активних заходів безпеки системи, так і на недостатню обізнаність або пильність співробітників [9].

Найбільш популярний методології тестування [8]:

- посібник з методології тестування безпеки з відкритим кодом; - Спеціальна публікація Національного інституту стандартів і технологій (NIST) 800-115;

- керівництво з тестування OWASP;

- стандарт виконання тестування на проникнення;

- структура оцінки безпеки інформаційних систем.

Для перевірки безпеки веб-додатків доцільніше використовувати методологію OWASP. Ця методологія заснована на методі чорного ящика – інформація про тестований додаток обмежена або відсутня взагалі. Безпека програмного забезпечення охоплює дуже широку область тем. Щоб мати безпечне програмне забезпечення, потрібно враховувати багато речей. На рисунку 2.1 ілюструються різні сфери безпеки, які вони покривають, і що необхідно враховувати.

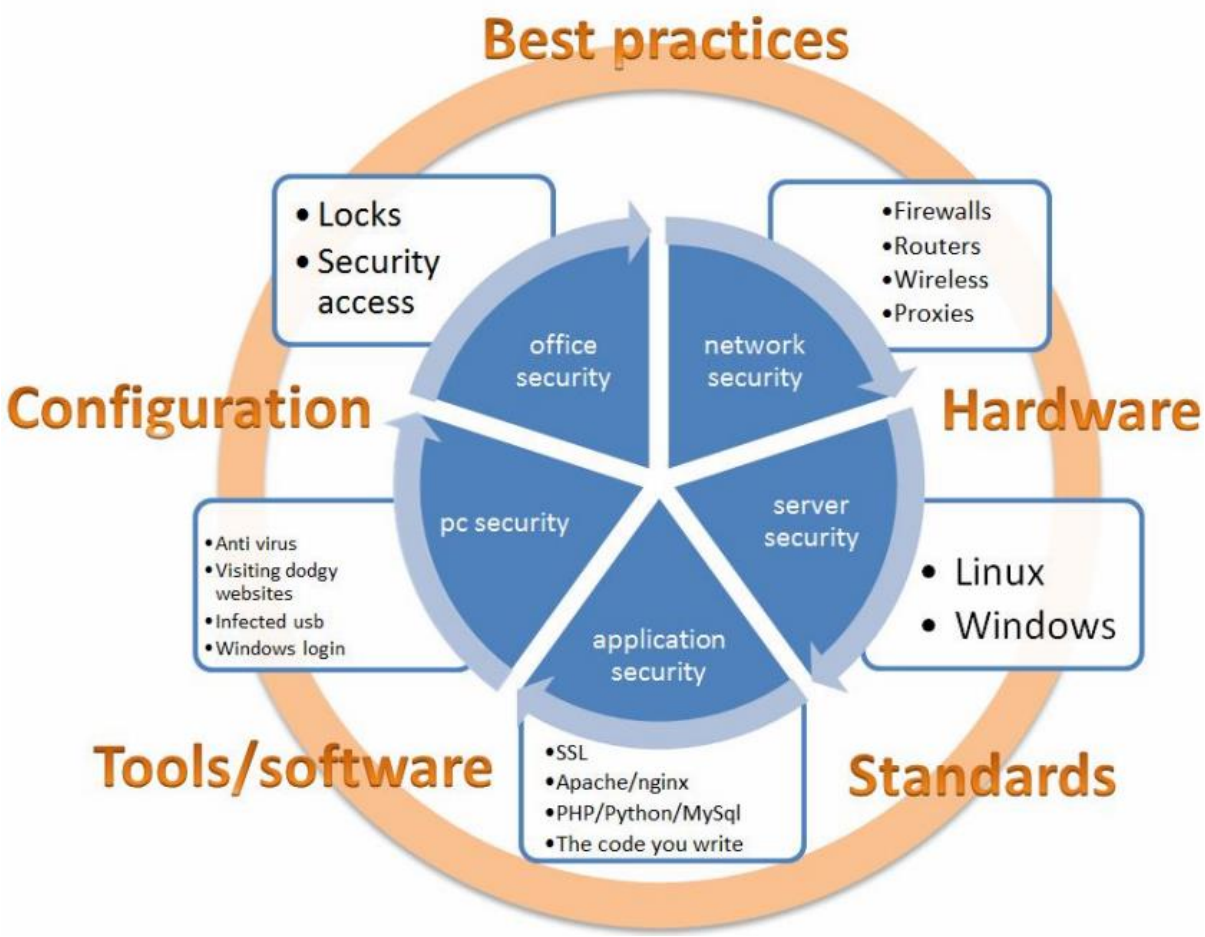

Рисунок 2.1 – Діаграма сфер безпеки веб-додатків

2.2 Найважливіші вразливості OWASP

Розуміння поширених вразливостей у веб-додатках допомагає краще підготуватися до захисту даних від таких атак. Завдяки знанням, отриманим від досліджень, користувачі та розробники можуть бути краще підготовлені для боротьби з найпоширенішими атаками та формувати рішення для запобігання майбутнім атакам на їхні веб-додатки. Вразливості існують у багатьох формах у сучасних веб-додатках, які можна легко пом’якшити, вкладаючи час та дослідження [10].

Нижче перелічено найважливіші вразливості 2021 року.

Атаки SQL Injection відбуваються, коли введені дані користувача не очищені належним чином і мають найбільший вплив на Інтернет. Коли зловмисники знаходять ін’єкцію SQL, у більшості випадків вони можуть отримати дані з бази даних, а в деяких випадках призводять до віддаленого виконання коду в цільовій системі. Атаки ін'єкцій SQL бувають у багатьох формах, таких як атаки на основі помилок, атаки на основі об'єднань, сліпі атаки та атаки поза діапазоном [10].

Порушена аутентифікація – це термін, який використовується для позначення кількох уразливостей, які дозволяють зловмисникам використовувати та видавати себе за користувачів веб-додатків. Існують різні методи, за допомогою яких зловмисники можуть отримати облікові дані користувача або зловити сеанси користувача, щоб мати можливість видавати себе за цих користувачів, наприклад, слабкі облікові дані користувача, які можна вгадати, неправильно збережені облікові дані, такі як паролі, які не були хешовані, які можуть бути вилучені з інших типів атак. такі як ін'єкції SQL, ідентифікатори сеансів, які відкриваються в URL-адресах, атаки фіксації сеансів, фіксовані ідентифікатори сеансів і особливо паролі, ідентифікатори сеансів та інші облікові дані, які надсилаються через незашифровані з'єднання, такі як HTTP [10].

Конфіденційні дані – це веб-програми, які не захищають таку інформацію, як паролі, фінансову інформацію або дані про стан здоров’я, що може призвести до того, що кіберзлочинці зловживають цією інформацією для отримання

несанкціонованого доступу до облікових записів користувачів, вчиняють шахрайські дії, такі як онлайн-покупки за допомогою вкраденого платежу. інформації або здійснювати шантаж отриманими конфіденційними даними. Розкриття конфіденційних даних може призвести до фінансових втрат, завдати шкоди репутації корпорацій, які розкрили свою інформацію чи активи, і спонукає підприємства оплачувати витрати на розслідування порушень даних. Захист від таких атак залежить від законодавчої бази країни та галузі, оскільки їх ігнорування може призвести до фінансово руйнівних результатів.

Атаки введення зовнішніх об’єктів XML, також відомі як ін’єкції XXE, відбуваються, коли зловмисники зловживають аналізаторами розширеної мови розмітки (XML) на веб-серверах, надсилаючи на веб-сервери спеціально створені шкідливі XML-документи, які обробляються і можуть призвести до відмови в обслуговуванні, віддаленого виконання коду або підробка запитів на стороні сервера.

Порушений контроль доступу складається з кількох можливих векторів атак, таких як обхід перевірок контролю доступу, редагування облікових записів інших користувачів, підвищення привілеїв, неправильні налаштування CORS, які дозволяють несанкціонований доступ до обмежених API, маніпулювання метаданими за допомогою маркерів контролю доступу, таких як веб-токени JSON (JWT) або доступ неавторизовані веб-сторінки як непривілейований користувач, що може призвести до контролю зловмисників бізнес-функцій або можливості отримання зловмисниками всіх даних. Рекомендується використовувати списки контролю доступу та забороняти доступ до функцій, використовуючи код на стороні сервера, де зловмисники не можуть отримати доступ до метаданих або контролювати їх.

Неправильна конфігурація безпеки відноситься до веб-програм, які були неправильно налаштовані таким чином, що залишають їх підданими загрозам безпеки. Вони можуть включати неправильні конфігурації брандмауера, відкриті порти адміністрування, які піддають програму віддаленим атакам, або застарілі програми, які намагаються спілкуватися з програмами, які більше не існують.

Забезпечення того, що конфігурації проходять належний процес забезпечення якості та ретельно перевіряються, тестуються та перевіряються, зменшує поверхню атаки від такого типу вразливості.

Міжсайтові сценарії, також відомі як XSS, мають багато форм, які призводять до різних результатів залежно від типу виконуваного XSS, але зазвичай відбувається, коли зловмисники вводять у веб-додаток шкідливі сценарії, які потім розкривають конфіденційну інформацію, внутрішні служби або розкривають файли cookie привілейованих користувачів.

Небезпечні атаки десеріалізації відбуваються, коли програми намагаються перетворити шкідливі дані, які контролює зловмисник, у внутрішні структури даних, які контролюються програмою. Впроваджуючи спеціально створені корисні навантаження, зловмисники можуть взяти під контроль змінні, функції та внутрішні стани програми. Це часто призводить до вразливостей віддаленого виконання коду, а також до впливу операційної системи веб-додатка.

Використання компонентів з відомими вразливими місцями означає використання певного програмного або апаратного забезпечення з відомими вразливими місцями, незалежно від того, чи вони були припинені або закінчилися терміном служби. Зловмисники, швидше за все, використовують поширені або відомі експлойти для отримання доступу до систем, а не виявляють нові вразливості. Захист від цієї вразливості вимагає відстеження залежностей програми, належної документації, видалення невикористаних залежностей, видалення мертвого коду та включення залежностей у політику оновлення програми, процедури та життєвий цикл обслуговування.

Недостатнє ведення журналів і моніторинг означає відсутність належних механізмів ведення журналів, які допомагають у моніторингу та виявленні інцидентів безпеки. Це дозволяє зловмисникам вести свою діяльність непомітно, що значно ускладнює завдання виявлення інцидентів та реагування на атаки. Журнали використовуються не лише для відстеження діяльності зловмисників або виявлення помилок та інших аномальних дій, які можуть мати місце в програмі.

Крім того, багато нормативних вимог залежать від належних механізмів ведення моніторингу.

2.3 Методи самозахисту веб-додатку під час виконання

Додатки соціальних мереж стали важливим джерелом Інтернет-послуг і трафіку. Як правило, він заснований на динамічних сторінкових програмах, таких як обмін інформацією, інтерактивний зворотний зв’язок і запити на послуги, що надаються веб-програмами. Більшість хмарних платформ та інтерфейсів керування мережевими пристроями також використовують веб-інтерфейси. Упорядковуючи додаткові оператори сценаріїв у введених користувачами, за відсутності надійних методів фільтрації, міжсайтові сценарії можуть викликати ефект вставки зайвого коду на сторінки веб-програм і порушення логіки вихідної сторінки, спричиняючи зловмисних дій клієнта.

Зазвичай уразливості міжсайтових сценаріїв можна уникнути за допомогою безпечних методів програмування, таких як фільтрація даних користувача. Оскільки веб-додатки побудовані на мові сценаріїв високого рівня, розробники більше стурбовані дизайном інтерфейсу з Web 2.0, а технології інтерфейсу стають більш складними. В результаті ускладняються перетікання змінних. Якщо це відсутність загального дизайну та розробки безпеки, а також немає ефективної фільтрації та обмежень, оскільки вразливості, засновані на архітектурі такого типу додатків, неминуче існуватимуть міжсайтові вразливості, які стануть актуальною проблемою безпеки мережі в умовах широкого застосування. Уразливості міжсайтових сценаріїв широко існують у веб-додатках у звіті про загрози безпеки OWASP десяти найбільших (OWASP top 10), ін’єкція команд і міжсайтове скриптування залишаються важливими загрозами [8]. Його можна виявити як на стороні сервера, так і на стороні клієнта. Метод виявлення на стороні сервера в основному полягає в тому, щоб переглянути введений користувачем вміст, відфільтрувати ненадійний вміст і зробити так, щоб шкідливий код сценарію не міг досягти браузера користувача. З точки зору

виявлення на стороні сервера, репрезентативними методами є вбудовані політики з підтримкою браузера (BEEP). Щоб запобігти введенню зловмисного коду сценарію у веб-програми, BEEP реалізує білий список довірених сценаріїв. Таким чином, створюється лише код надійного сценарію веб-розробником може виконуватися на клієнті, а решта відфільтровується. Однак BEEP має серйозні недоліки при роботі з динамічно генерованими сценаріями. Особливо коли поточна структура веб-моделі дозволяє процедурний дизайн, проблема стає більш очевидною, і були складніші атаки XSS, спеціально спрямовані на BEEP.

Метод визначення політики безпеки вмісту (Content Security Policy - CSP) довіряє виконанню браузера. Вважається, що причиною проблеми XSS є недоліки у веб-додатку. З цієї причини CSP додає багато атрибутів до основної частини кожної сторінки, створеної на стороні сервера. Під час створення сторінки надійність сценаріїв, зображень або іншого вмісту на сторінці визначається цими атрибутами. Таким чином, він забезпечує функцію автоматичного захисту від несправності. Оскільки ця політика поширюється на всю сторінку, ненадійний вміст обмежується цією політикою. Однак, оскільки CSP накладає суворі обмеження на створені сторінки, це спричинить серйозні проблеми з продуктивністю перед великомасштабними веб-додатками, а також не може добре адаптуватися до веб-архітектури дизайну процедурної моделі [11].

WAF (брандмауер веб-додатків) проникає у веб-протокол для фільтрації та використовує технологію брандмауера для обмеження з'єднань. WAF зазвичай розгортається на передньому кінці кластера веб-серверів для захисту веб-сайтів і використовує режим зворотного проксі для виконання двонаправленої фільтрації пакетів запитів HTTP. Якщо брандмауер розуміє протокол і семантику параметрів програми, це призведе до більш точного виявлення вразливостей. Однак із збільшенням складності додатків і швидкою появою нових типів програм і технологій (таких як JSON, REST тощо), здатність WAF забезпечувати точне виявлення вразливостей без штучної настройки ще не реалізована. Сьогодні мало хто вважає, що WAF розгорнуто в режимі безумовного блокування для будь-якої програми, що доводить притаманну неточність технології WAF. Деякі

вдосконалені методи використовують машинне навчання для вивчення правил підпису, а також стикаються з труднощами отримання набору даних і робочого навантаження на тегування вручну [11]. Комбінація методів виявлення на стороні сервера і клієнта в основному полягає в тому, що клієнтський браузер повинен проаналізувати повернутий HTML-документ способом, визначеним сервером, щоб шкідливий код не міг бути виконаний, навіть якщо він досягне клієнта. Створення наскрізного довіреного шляху між користувачем веб-програми та веб-сервером. Типовими прикладами є цілісність структури документів (DSI) [11].

Технологія RASP (Runtime application self-protection) є вдосконаленням WAF. Вона вводить код захисту в прикладну програму, інтегрується з прикладною програмою, а також відстежує та блокує атаки в режимі реального часу, так що програма має власні можливості захисту. Захищеному додатку не потрібно вносити жодних змін у кодування, лише за допомогою простої конфігурації. Цей метод підходить для рішень динамічного захисту від виправлень для певних вразливостей. Вона поки не може самостійно та комплексно вирішити певний тип загрози безпеці [11].

2.4 Техніка виявлення атак з використанням мови структурованих запитів

Більшість методів виявлення атак – це методи, які використовуються веб-розробниками для виявлення атак, як правило, під час виконання коду JavaScript на стороні клієнта. Сьогодні існує багато методів, і більшість з них відрізняються один від одного у виявленні та запобіганні SQLIA за допомогою спеціальних інструментів і механізмів кодування для виявлення або запобігання зловмисних атак SQL на цільову базу даних на сервері. Перш ніж представити техніку виявлення, важливо показати дуже поширені типи вразливостей безпеки [12].

Тип I - перевірка введених даних – це спроба перевірити або відстежити будь-які підозрілі вхідні дані на предмет можливої зловмисної поведінки. Неправильна перевірка поля даних може призвести до багаторазового виконання шкідливого коду без належної та точної перевірки початкового наміру. Таким

чином, зловмисник, використовуючи SQL, може захотіти скористатися неправильною перевіркою, щоб він міг виконати шкідливий код, а потім здійснити атаку на цільову базу даних.

Тип II - відсутність поділу між типами даних, які приймаються як вхідні.

Тип III - будь-яка затримка процесів до етапу виконання, оскільки наявні змінні вимірюються, незважаючи на те, що вихідний код використовує вираз для здійснення атаки.

Деякі статті в літературі навіть посилаються на збережені процедури як на засіб проти SQLIA. Оскільки збережені процедури знаходяться на передньому плані бази даних, запропоновані ними методи не можна застосовувати для захисту самих збережених процедур. Наприклад, деякі дослідження запропонували нову техніку захисту від атак, спрямованих на збережені процедури. Ця техніка поєднує статичний аналіз коду програми з перевіркою під час виконання, щоб виключити такі атаки. Інші дослідники розробили метод, який виявляє та запобігає атакам SQL-ін’єкції, перевіряючи, чи введені користувачем зміни в результатах запиту. Вони запропонували метод виявлення атак ін'єкції SQL за допомогою токенізації запитів, який реалізується методом Query Parser, який використовує метод Black Box Testing, для тестування веб-додатків на наявність вразливостей ін'єкції SQL. Ця техніка використовує веб-сканер для визначення всіх точок у веб-додатку, які можна використовувати для введення SQLIA. Інші дослідники використовували техніку під назвою динамічний аналіз, де це дуже корисно для проведення аналізу часу виконання або динамічного SQL-запиту, він генерується за допомогою даних, що вводяться користувачем, спочатку шляхом реалізації веб-додатка. Техніка виявлення також у постгенерованій категорії може виконувати запит перед відправкою запиту до цільової бази даних на стороні сервера. Існує ще одна запропонована методика під назвою SQL-IDS, це підхід на основі специфікації для виявлення шкідливих вторгнень. Деякі дослідники SQLIA пропонують використовувати нову методологію, засновану на специфікації, а також на характеристиках експлуатації та виявлення шкідливих ін’єкцій SQL, щоб уникнути вразливостей. Виявлено, що

виявлення, засноване на певному запиті, дозволяє системі виконувати аналіз, зосереджений на незначних обчисленнях без будь-якої необхідності створення хибно-негативних або хибно-позитивних результатів. У центрі уваги дослідження методів виявлення I типу, запропонована методика виявлення, яка називається Combined Detect, заснована на JavaScript, і два рішення, де перевірка в залежності від введених даних як спроба довести або відфільтрувати будь-який підозрілий вхід містить специфікацію зловмисної поведінки. Більшість методів показали, що прийнята методика виявлення недостатньої перевірки введення дозволить виконувати шкідливий код без належної перевірки його наміру. Тому будь-яка методика виявлення повинна перешкоджати зловмисникові скористатися перевагами недостатньої перевірки введення, які загрожують цільовій базі даних і допомагають зловмиснику використовувати шкідливий код SQL для легкого проведення атак.

В даному розділі виконано огляд теоретичних основ захисту веб-додатків.

Безпеці веб-додатків необхідно приділяти дуже багато уваги. Безпека веб-додатків залежить від якості програмного коду, від кваліфікації системного адміністратора та від компетенцій усіх користувачів, які мають доступ до чутливої інформації.

Вразливості OWASP є дуже шкідливими і в цьому списку зосереджені найнебезпечніші вразливості, які можуть коштувати купу грошей, або підрив ділової репутації, чи довести аж до втрати бізнесу.

Методи самозахисту веб-додатку під час виконання мають деякі складнощі з виявленням вразливостей без штучного налаштування, але з використанням машинного навчання можуть бути корисними у нагоді.

Техніки виявлення атак з використанням мови структурованих запитів є необхідними, однак фільтрація даних здається дуже ефективною та найпростішою технікою. Незважаючи на те, що виявлено, що методи виявлення SQLIA, які використовують перевірку введення, схильні до великої кількості хибно-позитивних результатів, і все ж немає 100% гарантії, що немає помилкових негативів.

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ЗАХИЩЕНОЇ АВТОМАТИЗОВАНОЇ ВЕБ-СИСТЕМИ ТЕСТУВАНЬ ЗНАНЬ

3.1 Засоби реалізації програмного продукту

Для побудови веб-додатку було використано мови програмування JavaScript [13-18] та PHP [19-24], фреймворки Nuxt [25-30] та Laravel [31-36]. Для розробки використовувалось інтегроване середовище PhpStorm.

Усі веб-сайти працюють на трьох компонентах: сервері, базі даних та клієнті.

Клієнт – це просто браузер, який людина використовує для перегляду сайту, і саме там клієнтська технологія розпаковується та обробляється.

Сервер знаходиться у віддаленому місці у будь-якій точці світу, що зберігає дані про місцезнаходження, керуючи внутрішньою архітектурою сайту, обробляючи запити та відправляючи сторінки до браузера. Клієнт знаходиться в будь-якому місці, де користувачі переглядають сайт: мобільні пристрої, ноутбуки чи настільні комп'ютери.

Серверний скрипт виконується веб-сервером, а сценарій клієнта виконується браузером.

JavaScript – це клієнтська скриптова мова програмування. Найбільш широко використовуваний клієнтський скрипт – майже кожен інтерфейс сайту – це поєднання JavaScript, HTML [37-42] та CSS [43-48].

Nuxt.js – це фреймворк для створення програм на Vue.js. Дозволяє створювати готові до роботи веб-програми і покликаний спростити розробку універсальних та односторінкових сервісів [25].

Переваги Nuxt.js:

- просте створення універсальних програм. Одне з головних переваг у тому, що фреймворк полегшує створення універсальних додатків. Останні написані на JavaScript, причому скрипти використовуються як на стороні клієнта, так і сервері;

- статичний рендеринг. Найбільша новація приходить з командою nuxt generate. Вона повністю створює статичну версію вашого сайту. Фреймворк створить HTML для кожного з маршрутів і помістить його у власний файл;
- автоматичне розбиття коду. Фреймворк може генерувати статичну версію вашого сайту із спеціальною конфігурацією Webpack. Для кожного статично генерованого маршруту (сторінки) він також отримує власний файл JavaScript, що містить тільки код, необхідний для запуску;
- відмінна структура проекту за промовчанням. У багатьох невеликих програмах Vue ви керуєте структурою коду, у кращому разі, у кількох файлах. Структура Nuxt.js за промовчанням дає відмінний старт для організації вашого сервісу в зрозумілій формі.

Абревіатура PHP розшифровується як “PHP: Hypertext Preprocessor” (PHP: Препроцесор Гіпертексту). PHP - це мова програмування загального призначення, що використовується на серверній стороні. Це означає, що коли відвідувач запитує сторінку сайту, сервер отримує запит, передає управління інтерпретатору PHP, він виконує всі необхідні операції, віддає готовий код серверу і потім сервер відправляє готову сторінку браузеру.

Laravel – фреймворк корпоративного рівня мовою програмування PHP, створений для розробки складних сайтів та веб-додатків [31].

Приклади на Laravel забезпечують більш високу продуктивність відповідно до додатків, створених за допомогою інших фреймворків. Це можливо в тому числі завдяки системі кешування. Драйвер файлового кешування зберігає безліч елементів у файловій системі. Це дозволяє швидко розробляти додатки.

Задля обміну між клієнтською та серверною частинами використовується REST - це концепція (архітектура) в організацію взаємодії між незалежними об'єктами (додатками) у вигляді протоколу HTTP. Включає набір принципів (рекомендацій) взаємодії клієнт-серверних додатків. Зазвичай вона представлена у форматі JSON.

Є кілька способів розробки веб-додатків, але на сьогоднішній день офіційний і найпопулярніший спосіб – це PhpStorm. Програмне забезпечення JetBrains PhpStorm є спеціалізованим засобом веб-розробки, орієнтованим на веб-додатки та іншими видами програм, які можна створювати за допомогою мови PHP та з використанням HTML, JavaScript та CSS. Рішення PhpStorm здійснює розгортання та синхронізацію проектів через протокол FTP. Середовище PhpStorm пропонує функції автоматичного завершення умовних конструкцій PHP у коді, інспектування коду, різні алгоритми рефакторингу та швидку навігацію за кодом.

Реалізований у PhpStorm графічний PHP-наладчик підтримує умовні точки зупинки, відстеження значень та автоматизований вхід у налагодження окремих процедур. Для тестування програм підтримується каркас тестових модулів PhpUnit та графічний інтерфейс для запуску тестів. При редагуванні коду виділяються конструкції синтаксису, здійснюється розширене форматування конфігурації, виявлення помилок у режимі реального часу та завершення коду. Редактор PhpStorm враховує коментарі до коду при його завершенні, автоматично вибираючи оптимальне вирішення проблеми. PHP-рефакторинг та редагування шаблонів гарантує зміну проекту в найкоротші терміни. PhpStorm дозволяє робити візуалізацію коду в ієрархічному вигляді та забезпечує швидку навігацію по всім елементам проекту.

3.2 Архітектура програмного продукту

Структура програмного продукту є модульною. Функціональна структура додатку включає в себе наступні модулі:

- модуль автентифікації та реєстрації;
- модуль відновлення паролю;
- модуль публікацій;
- модуль тестувань знань;
- модуль груп;

- модуль користувачів;
- модуль управління особистою інформацією.

Модуль автентифікації та реєстрації дозволяє ідентифікувати користувачів по їх електронній адресі. Без цього модулю користувач не може бути ідентифікований, а це значить, що він не може використовувати функціонал додатку. Цей модуль, використовуючи API віддаленого серверу, відправляє запит на перевірку або створення нового облікового запису.

В успішному випадку сервер віддає токен, тобто деякий унікальний ключ. Унікальний ключ складається з 512 бітів інформації, тобто 64 символи по одному байту, що на практиці унеможливлює підбір цього ключа зловмисником. За допомогою унікального ключа користувач може здійснювати керування обліковим записом та отримувати інформацію з сервера, де необхідна автентифікація за токеном.

Модуль відновлення паролю дозволяє відновлювати пароль, якщо виникає така ситуація, коли користувач не пам’ятає пароль від облікового запису. На вказану електрону адресу посилається лист з токеном для відновлення паролю. Використовуючи цей токен, користувач може змінити пароль, але на протязі деякого часу, так як токен обмежений за часом.

Модуль публікацій дозволяю створювати публікації, переглядати та видаляти їх. Модуль є повністю незалежним від інших, тому з легкістю може переноситись на інші веб-додатки.

Модуль тестувань знань тестувань знань дозволяє створювати тестування з різними питаннями та відповідями, які можуть підтримувати математичні формули. Модуль підтримує перемішування питань та відповідей в питаннях. Є можливість завдання обмеження за часом, щоб проходження тестування було доступно лише в деяких проміжках часу. Також він дозволяє проходити тестування, переглядати вже пройдені тестування та бачити результати тестувань.

Модуль груп дозволяє створювати та видаляти групи, можливість додавати користувачів до груп. При використанні модуля груп надається можливість чітко

визначати область видимості для тестувань знань, тобто надавати доступ до тестування лише для вказаних груп.

Модуль користувачів надає можливість перегляди усіх користувачів в системі, обмежувати або надавати доступ до системи користувачу, переглядати дії користувача. Є можливість змінювати профіль користувача, а саме змінювати ім’я, групу, роль, пароль. Використовуючи фільтр, надається можливість знаходити лише певних користувачів, які задовольняють умовам фільтру.

Модуль управління власною інформацією надає можливість лише змінювати пароль та ім’я в системі для усіх ролей в системі.

3.3 Технологія реалізації програмного продукту

Під час реалізації веб-додатку було виявлено чимало проблем як з клієнтської частиною, так із серверною.

Першою проблемою, що було виявлено в ході розробки, стало те, що додаток використовує фреймворк Nuxt, який в свою чергу використовує NodeJS збоку серверу. Майже всі додатки, що спершу роблять відображення контенту зі сторони серверу, можуть мати проблеми з витіком пам’яті. Тобто це процес, при якому відбувається постійне зменшення доступної програмі оперативної пам'яті, причому програма не має інформації про більшу частину зайнятої пам'яті. При витіку пам’яті додаток рано чи пізно переходить до аварійного завершення, що означає, що користувачі можуть бачити помилку 502 з боку серверу та не можуть користуватися веб-додаток. Проблема була вирішена використанням PM2, що є менеджером виробничих процесів для додатків Node.js з вбудованим балансувальником навантаження. Це дозволяє підтримувати додатки назавжди, перезавантажувати їх без простоїв і полегшувати звичайні задачі системного адміністратора. Менеджер слідкує за станом веб-додатку, тому при або аварійному завершенні, або при витіку пам’яті за деяку межу, або при раптовому перезавантаженню серверу, веб-додаток буде перезавантажений.

Другою проблемою виявилося те, що коли користувач намагається відновити пароль та просить сервер, щоб він відправив листа до заданої електронної пошти, то ця операція відбувається синхронно, що є проблемою з точки зору великого відгуку від серверу. Вирішенням цієї проблеми було використано менеджер фонових завдань. Створювати завдання у чергу, які можуть опрацьовуватися у фоновому режимі. Переміщаючи трудомісткі завдання в чергу і виконуючи їх у фоні, програма може швидше обробляти веб-запити та швидше відповідати клієнту. Але в даному випадку, фонові завдання повинні оброблятися зі сторони автоматичного програмного забезпечення без відвідання будь-якої сторінки для їх запуску.

Автоматична обробка фонових завдань передбачає, що програма, яка оброблює завдання, також може потерпіти невдачу та аварійно завершитись. Отож для вирішення цієї проблеми було використано supervisor - клієнт-серверну систему, яка дозволяє контролювати низку процесів у операційних системах, подібних до UNIX.

Третьою проблемою є дуже повільна віддача статичного контенту веб-додатком, адже Nuxt призначен в першу чергу для візуалізацію з JavaScript коду на HTML представлення. Отож Nuxt дуже погано підходить в якості веб-серверу, тому було вирішено використовувати в якості прошарки NGINX, що є HTTP-сервером і може виконувати завдання зворотного проксі-серверу. Таким чином, коли користувач вперше потрапляє на сторінку, nginx оброблює запит та делегує завдання візуалізації через проксі-сервер до Nuxt. Натомість вся статика, яка буде проходити на сервер, буде оброблятися саме веб-сервером NGINX, що підвищує продуктивність серверу та економить ресурси серверу.

У відповідності до перелічених проблем узагальнена схема запитів до серверу зображена на рисунку 3.1.

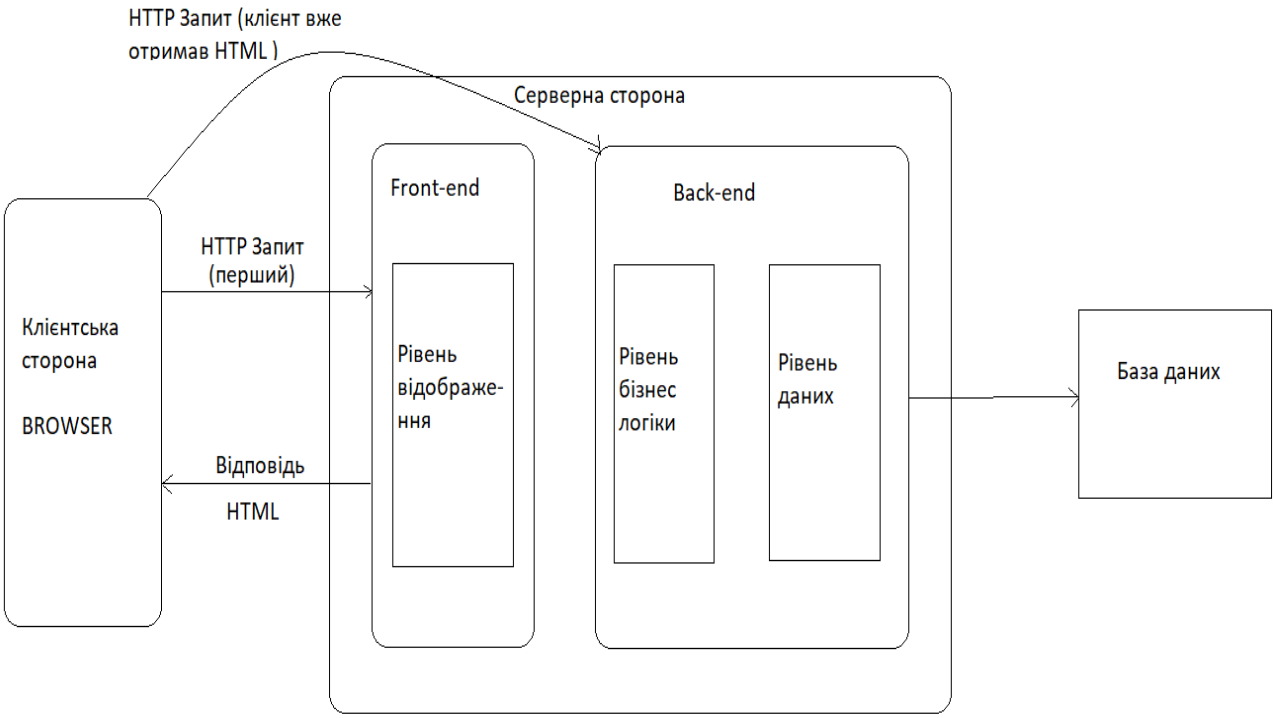

Рисунок 3.1. – Схема запитів до серверу с використанням візуалізації на стороні серверу

3.4 Інструкція з експлуатації програмного продукту

Інтерфейс веб-додатку переведений на дві мови: українську та англійську. Щоб змінити мову, необхідно натиснути на мову та у спадаючому меню обрати бажану мову.

Будь-яке повідомлення, що буде отримано від серверу, буде відображено у верхньому лівому кутку.

При відкритті веб-додатку користувач бачить екран для проходження автентифікації. Для проходження автентифікації користувачу необхідно заповнити обов’язкові поля: «електронна пошта» та «пароль». Далі необхідно натиснути на кнопку «Увійти». У випадку, якщо заданої пари електронна пошта/пароль не буде знайдено, система сповістить користувача у вигляді повідомлення.

Будь-який студент або викладач, що бажає зареєструватися у системі, має натиснути на посилання «Створити», що знаходиться знизу форми автентифікації користувача. Після переходу на посилання, користувач бачить поля: «Ім’я», «Електронна пошта», «Пароль», «Підтвердження паролю», «Роль». Всі вони є обов’язковими для заповнення. На деякі поля накладені обмеження по мінімальній або максимальній довжині рядків. Під полями, що не будуть задовольняти умовам клієнтської валідації, буде відображена помилка. Також необхідно привернути увагу до спадаючого списку ролей у системі. У тому випадку, якщо буде обрано значення «Студент», користувач побачить ще одне обов’язкове поле «Група». За замовчуванням у системі створена група «Common», кожна інша група самостійно створюється адміністратором у системі. Після введення усіх полів необхідно натиснути на кнопку «Зареєструватися». У разі успіх користувач потрапить до сторінки автентифікації.

Може виникнути ситуація, коли користувач вже є зареєстрованим у системі, але по деякій причині він забув пароль. Для відправлення посилання на електронну пошту для скидання паролю, користувачу необхідно натиснути на посилання «Натисніть, щоб відновити пароль», що знаходиться від кнопкою «Увійти» на сторінці автентифікації та заповнити поле електронної пошти і натиснути на кнопку «Далі». Після отримання листа з посиланням на скидання паролю, необхідно перейти по посиланню. По відкритому посиланню користувач побачить два обов’язкових поля «Пароль» та «Підтвердження паролю», які необхідно буде заповнити та натиснути кнопку «Змінити пароль». Після зміни паролю, користувачу буде запропоновано увійти у систему.

Як було зазначено раніше, у системі існує три ролі: «Адміністратор», «Викладач» та «Студент»:

- адміністратор має усі повноваження у системі, а саме: створення та видалення публікацій і груп; створення тестувань, видалення та проходження усіх раніше створених тестувань у системі; змінювати всю інформацію користувача, у том числі змінювати групу, роль, пароль та підтверджувати у системі; змінювати власний профіль;

- викладач має повноваження на створення тестувань та проходження тих тестувань, котрі є створеними або у загальнодоступних; перегляд користувачів; змінювати власний профіль;
- студент має повноваження лише на проходження тестувань, що є загальнодоступними або відносяться до групи, до якої належить сам студент.

Після проходження автентифікації, стає доступна панель користувача. В залежності від повноважень, користувач бачить доступний йому функціонал системи зліва у меню. Меню містить наступні пункти: «Головна», «Тестування», «Користувачі», «Групи», «Публікації», «Налаштування» та «Вийти», що зображені на рисунку 3.2.

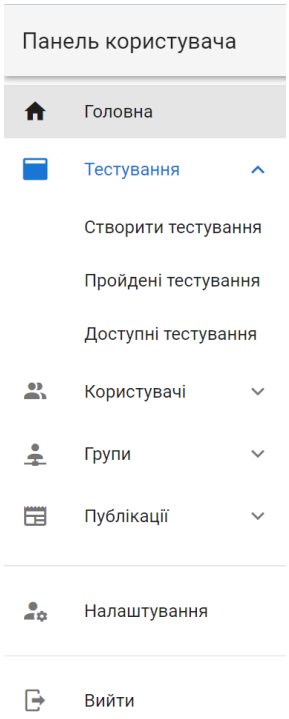

Рисунок 3.2. – Головне меню в панелі користувача

Щоб створити публікацію, необхідно натиснути на пункт головного меню «Публікації», натиснути на посилання «Створення публікації», заповнити поля «Титул» та «Зміст» і натиснути на кнопку «Створити новину». Перегляд створених та управління ними здійснюється по натиску на посилання

«Управління», де можна побачити усю необхідну інформацію та видалити будь-яку, якщо є така необхідність.

Створення груп відбувається наступним чином. Необхідно натиснути на пункт «Групи» в меню, перейти по посиланню «Створення групи» та заповнити ім’я групи. Управління групами здійснюється по переходу на посилання «Управління» для підпункту «Групи».

Для перегляду всіх користувачів, студентів та викладачів необхідно натиснути на пункт меню «Користувачі» та обрати необхідний підпункт. По всім користувачам перелічена інформація щодо електронної пошти, ролі, групи та стану. Так як кожен користувач, що був зареєстрований, має бути спершу підтверджений адміністратором системи, то у полі «Дії» можна або підтвердити користувача, або відхилити йому доступ до системи. У разі необхідності використання фільтру по переліченим вище полям, треба натиснути на панель пошуку та ввести дані, яким повинен задовільнити фільтр.

По натисненню на ім’я користувача, можна переглянути його дії щодо автентифікації в систему, виходу з системи, спробу початку проходження тестування або його завершення. По всім діям записується IP адреса. Якщо необхідно відредагувати дані користувача, то по натисненню на «Редагування користувача» можна змінити бажану інформацію.

Для створення тестування, перегляду пройдених тестувань або задля проходження тестування, треба натиснути на пункт «Тестування» та обрати необхідний підпункт.

На сторінці «пройдені тестування» відображається інформація щодо коли було пройдено, титул тестування та кількість отриманих балів.

На сторінці «доступні тестування» розташована інформація щодо ким було створено тестування, область видимості, титул, опис, обмеження за часом, кількість пройдених разів тестування користувачами системи. Проходження або видалення тестування відбувається по натисненню на кнопку «Дії» та обранням бажаного підпункту.

На сторінці «створення тестування» необхідно вказати обов’язкові поля «Титул», «Опис», «Область видимості». Поля «Початок в» та «Закінчення в» є необов’язковими, проте при їх заповнені вони повинні задовольняти умовам, що обидва поля є майбутніми датами та поле «Закінчення в» випереджає поле «Починається в». Область видимості може бути як публічна, так і бути доступною лише для деяких груп користувачів.

Надалі необхідно додати мінімум одне питання. Максимальна кількість питань обмежена до тридцяти питань.

Щоб додати питання, необхідно натиснути на кнопку «Додати питання». У випадку, якщо необхідно видалити обране питання, то треба натиснути на кнопку «Видалити питання». Питання має обов’язкові поля «Тип», «титул» питання, «кількість балів» та необов’язкове поле «Зображення». Тип питання може бути з однією відповіддю або з декількома.

Кількість балів має бути більше, ніж 0. Аби додати відповіді до запитання, треба натиснути на кнопку «Додати відповідь», далі вказати титул відповіді та серед створених відповідей обрати правильні відповіді.

Титул питання та відповідей підтримує синтаксис LaTeX, тобто можна вказати математичні формули у форматі \(математична формула\).

Після заповнення всіх даних, натиснути на кнопку «Створити тестування». Якщо помилок не буде виявлено, користувачу буде відображена сторінка з доступними тестуваннями.

В даному розділі представлена практична реалізація захищеної автоматизованої веб-системи тестувань знань.

Детально описано вибір засобів реалізації програмного продукту: особливості мов програмування JavaScript та PHP, фреймворків Nuxt та Laravel, інтегрованого середовища розробки PHPStorm.

Розглянута архітектура програмного продукту «SaveTestingKnowledge», яка включає в себе наступні модулі: модуль автентифікації та реєстрації; модуль публікацій; модуль груп; модуль тестування знань; модуль управління користувачами; модуль управління особистою інформацією.

Представлені дві технології, що допомогли в вирішенні проблем, які виникли при розробці програмного продукту «SaveTestingKnowledge» при взаємодії між клієнтською та серверною частинами.

Наведені основні складові програмного продукту та інструкція з експлуатації.

4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Аналіз умов праці і вибір заходів і засобів захисту від небезпечних і шкідливих виробничих факторів

Сучасне суспільство має всі ознаки постіндустріального, або інформаційного, в якому обчислювальна техніка є типовим робочим інструментом. Її використання дозволяє зберігати, отримувати і обробляти інформацію, проектувати об’єкти і технологічні процеси, вирішувати творчі завдання тощо. Фактично більшість видів виробничої і невиробничої діяльності сучасної людини нині так чи інакше пов'язана із використанням персональних комп’ютерів або великих обчислювальних машин. Магістерська робота присвячена розробці захищеної веб-системи на основі оцінки знань. Системи подібного типу створюються спеціалістами з інформаційної безпеки, чиїм основним робочим обладнанням є саме персональний комп’ютер, підключений до мережі Інтернет, обладнаний потрібними периферійними пристроями. Обчислювальна техніка істотно впливає на здоров'я користувачів. Постановка і розв’язання завдань, пов'язаних із безпекою користувача обчислювальної техніки під час виконання трудових обов’язків, вимагає аналізу умов праці і визначення шкідливих і небезпечних впливів, а також аналізу фізіологічних реакцій організму на ці впливи.

Основними джерелами небезпечних і шкідливих виробничих факторів, що впливають на організм користувача ПК, є наступні елементи: системний блок, монітор, клавіатура, миша, колонки, модем, принтери. Окрім них, існують і інші джерела виробничих небезпек і шкідливих впливів, які знаходяться в робочому приміщенні, а також джерела зовні робочого приміщення – в самій будівлі і навколо неї. Зовнішніми джерелами виробничих небезпек і шкідливих впливів у виробничому приміщенні є персональні комп’ютери інших працівників, елементи електромережі, системи природного і штучного освітлення, системи опалення, вентиляції і кондиціонування повітря, будівельні матеріали і покриття стін, підлоги і стелі виробничого приміщення. Зовнішніми джерелами виробничих

небезпек і шкідливих впливів за межами виробничого приміщення можуть бути технічні системи і об’єкти, які визначають концентрацію шкідливих речовин у повітрі, що надходить у виробниче приміщення, рівень зовнішнього шуму або вібрації, рівень неіонізуючих випромінювань радіодіапазону.

Аналізуючи умови праці користувача ПК згідно із нормативними документами [49], виділяємо наступні небезпечні і шкідливі виробничі фактори:

- енергетичні впливи, а саме: електромагнітне випромінювання різної частоти, випромінювання видимої частини спектра;
- небезпека дії електричного струму промислової частоти;
- небезпека впливу статичної електрики;
- підвищена концентрація пилу в повітряному середовищі робочого приміщення;
- підвищений рівень шуму в робочому приміщенні;
- шкідливі фактори ергономічного характеру, які визначаються мірою відповідності елементів робочого місця ергономічним критеріям взаємодії користувача ПК із засобами праці в межах утвореної ними системи «людина – машина»;
- небезпеки надзвичайних ситуацій природного і техногенного походження.

Наслідки впливу електромагнітного випромінювання на організм людини залежать від довжини хвилі, інтенсивності випромінювання, тривалості опромінення, особливостей окремих біологічних тканин. Наприклад, випромінювання міліметрового діапазону поглинаються поверхневими тканинами – шкірою, випромінювання сантиметрового діапазону впливає на шкіру і підшкірні шари тканин, випромінювання дециметрового діапазону проникає в тіло на глибину 8 – 10 см. Реакції організму на вплив електромагнітного випромінювання поділяють на теплові та нетеплові (специфічні) – а саме

біохімічні зміни у клітинах та тканинах. Результатом нетеплових реакцій є порушення умовно-рефлекторної діяльності нервової системи, особливо мозку, порушення діяльності ендокринної системи, уповільнення кровотоку в серцевому м’язі.

Основним заходом захисту від енергетичних впливів є зменшення рівня опромінювання на робочих місцях. Показники опромінення мають відповідати нормативним документам, а саме ГОСТ 12.1.006 – 84 [50]. Також необхідно використовувати захист відстанню – розміщення робочих місць на можливому віддаленні від джерел випромінення – а також планувальні рішення, тобто таке розміщення робочих місць, яке мінімізує опромінення. В якості профілактичног заходу слід впроваджувати медичні огляди при вступі на роботу та періодичні медичні огляди не рідше ніж 1 раз на рік.

Під час роботи персонального комп’ютера на екрані монітора накопичується електричний заряд і створюється електростатичне поле. Те ж саме відбувається під час роботи лазерного принтера. Внаслідок того на екрані і принтері відбувається інтенсивне накопичення часток пилу розміром від 5 до 50 мкм. Напруженість створюваного обладнанням електростатичного поля може перебувати в межах від 8 до 75 кВ/м. При цьому на тілі користувача ПК створюється електростатичний потенціал величиною від - 3 до + 5 кВ. Як наслідок, частинки пилу можуть осаджуватися на обличчі та відкритій поверхні рук, також пил потрапляє в дихальні шляхи користувача ПК під час роботи. За результатами досліджень встановлено, що дія електростатичного поля напруженістю 15 кВ/м за проміжок часу, що перевищує 1 годину, є причиною виникнення процесів збудження в центральній нервовій системі, яке з часом проявляється у підвищеній втомлюваності, фізичному виснаженні, непереносимості гучних звуків і яскравого світла. Внаслідок накопичення пилу на шкірі можуть виникати шкірні захворювання – дерматит або екзема.

Користувач ПК може піддатися впливу електричного струму промислової частоти. Причинами цього можуть бути [51]:

- пошкодження ізоляції струмоведучих частин обладнання;

- порушення вимог підключення професійного обладнання до електричної мережі;
- використання електрообладнання, яке не потрібно при виконання трудових обов'язків (наприклад, електронагрівального).

Наслідками дії електричного струму є електричні удари – загальні ураження організму із особливим впливом на центральну нервову систему – і місцеві електричні травми, основними з яких є опіки.

Шкідливим виробничим фактором на робочому місці є підвищений рівень шуму у виробничому приміщенні із персональними комп'ютерами. Шум вважається загально-біологічним подразником, оскільки він негативно впливає не тільки на слуховий апарат людини, але викликає розлади в роботі серцево-судинної та нервової систем, сприяючи виникненню гіпертонічної хвороби. Крім того, шум може бути однією з причин виробничої втоми.

Значною особливістю роботи людини з персональним комп'ютером є принципово інший спосіб читання інформації. Під час роботи із монітором очі тривалий час працюють в незвичному для них режимі.

Безперервні або тривалі зорові навантаження викликають порушення функціонального стану зорових аналізаторів, які полягають у зниженні гостроти зору і акомодації, розвиненні короткозорості, зниженні контрастної чутливості зору і інших функціональних порушеннях зорового апарату.

Аналізуючи шкідливі виробничі фактори, слід приділяти уваги навантаженням на користувача ПК як на елемент системи «людина – машина». Під час праці в положенні сидячи виникають статичні та динамічні навантаження на опорно-рухову систему організму, наслідком яких стають біль у спині та шиї, біль у зап'ястях, стенокардія. Можливий розвиток м'язової слабкості, що призводить до зміни форми хребта. Статична поза під час роботи і нераціональна організація робочого місця можуть призводити до розладів опорно-рухової системи, наприклад, до шийного остеохондрозу, супроводжуваного

офтальмологічними порушеннями. Під час набору інформації з клавіатури ПК навантаження на кисті рук викликає запальні процеси в тканинах сухожилля.

Поєднання несприятливих факторів виробничого середовища (підвищений рівень шуму, недостатня освітленість, загальна втома) може призводити до нещасних випадків і виробничого травматизму. Тому організація безпечних умов праці є важливою вимогою для всіх підприємств і установ [52].

4.2 Аналіз техногенних небезпек і вибір заходів і засобів забезпечення безпеки у надзвичайних ситуаціях

Техногенна небезпека – потенційна властивість технічних об’єктів і систем завдавати шкоди і створювати загрози життю і здоров’ю людей, біосфері, іншим технічним об’єктам і елементам інфраструктури.

Найбільш реальна техногенна небезпека в будівлях адміністративного типу, в яких зазвичай розміщають робочі місця спеціалістів з інформаційної безпеки – це займання із подальшою пожежею.

Основні причини займань і пожеж:

- короткі замикання внаслідок перевантаження електричної мережі, пошкодження ізоляції проводів, порушення правил експлуатації обладнання;
- використання несправного електроустаткування: комп’ютерної техніки із пошкодженою ізоляцією, розеток, освітлювальних приладів;
- застосування обігрівачів відкритого типу в приміщеннях, де зберігається велика кількість паперових документів, оздоблення приміщень і меблі виготовлені з горючих і легкозаймистих матеріалів;
- куріння у недозволених місцях;
- неправильне поводження з обладнанням або з пожежо-вибухонебезпечними речовинами;

- використання піротехнічних пристроїв;
- навмисні підпали.

У випадку пожежі користувач ПК може піддаватися небезпечному впливу відкритого полум’я, підвищеної температури повітря і предметів, токсичній дії продуктів горіння (вуглекислого і чадного газу), токсичних продуктів термічного розкладання пластиків.

Тому оптимальним є впровадження профілактичних заходів пожежної безпеки, до яких належать:

- регулярний контроль стану електрообладнання і проводів електромережі;
- використання лише справної комп’ютерної техніки і периферійних пристроїв, заборона використання обладнання із пошкодженнями проводів, корпусів і іншими дефектами;
- контроль дотримання правил електробезпеки при експлуатації обладнання, заборона перевантаження електромережі;
- заборона використання опалювальних пристроїв і нагрівачів відкритого типу;
- дотримання режимів поводження із відкритим полум’ям.

Електричну мережу в приміщеннях слід виконувати відповідно до вимог ПУЕ для пожежонебезпечних зон та установок класів II–Ia. Прокладати кабелі через перекриття, стіни і фальшпідлоги необхідно в сталевих трубах з ущільненням з негорючих матеріалів.

Мережі аварійного електроживлення і освітлення, дистанційного та автоматичного пуску протипожежних систем та сигналізації слід прокладати окремо від силових та інших електричних мереж, а при спільній прокладці розділяти перегородками з негорючих матеріалів.

Система електроживлення ЕОМ повинна мати блокування, що забезпечує відключення її у разі зупинки системи охолодження та кондиціювання.

В адміністративних будівлях, або в будівлях із приміщеннями офісного типу обов'язково слід розміщувати на помітних місцях схеми евакуації персоналу і відвідувачів, забороняється захаращувати евакуаційні шляхи і закривати евакуаційні виходи. Усі працівники повинні регулярно проходити навчання, інструктажі, тренінги з пожежної безпеки та додержуватись під час праці вимог пожежної безпеки.

Оздоблення приміщень із робочими місцями спеціалістів з інформаційної безпеки слід виконувати з негорючих або важкогорючих матеріалів, дерев'яні конструкції рекомендовано обробляти антипіренами.

В якості засобів первинного пожежогасіння слід використовувати ручні або пересувні вогнегасники: порошкові (ПСБ, ПФ, ВП) або вуглекислотні (ОУ-2, ОУ-5). У замкнутих приміщеннях об'ємом до 50 $м^3$ замість ручних або пересувних вогнегасників можна використовувати підвісні порошкові вогнегасники, що спрацьовують автоматично (ОСП). Вогнегасник ОСП являє собою герметичний скляний сосуд, заповнений вогнегасним порошком, розміщений в корпусі. При досягненні температури повітря 100 °С або 200 °С сосуд вогнегасника автоматично розкривається і розсипає порошок навколо. При використанні ОСП у ручному режимі слід розбити «носик» вогнегасника і різким рухом засипати вогнище вогнегасним порошком. У приміщеннях об'ємом більше 50 $м^3$ вогнегасники ОСП рекомендується застосовувати для захисту найважливіших об'єктів.

Для забезпечення швидкої безпечної евакуації персоналу біля дверних прорізів, вимикачів, рубильників, на евакуаційних шляхах, для легкого виявлення шаф з первинними засобами пожежогасіння, слід використовувати фотолюмінесцентні евакуаційні стрічки та знаки евакуації, які здатні світитися протягом 30 хвилин. Такої тривалості світіння достатньо для евакуації з пожежонебезпечної зони.

4.3 Дослідження біоритмів спеціаліста з інформаційної безпеки

Біоритми людини є значущою складовою для визначення впливу психофізіологічних виробничих факторів на трудову діяльність. Вплив останніх може посилюватися або послаблюватися в залежності від фізичного і емоційного стану працівника. Відсутність збігу біоритмів із ритмами трудової діяльності підвищує ризик виникнення депресії, захворювань серця, діабету, ожиріння. Мотивація, працездатність, продуктивність праці значною мірою залежать від біоритмів людини. Для їх підвищення бажано знати власні фізичні, інтелектуальні і емоційні біоритми і розраховувати їх за стандартними методиками.

Вихідними даними для розрахунку є дата народження і задана дата розрахунку. Результатом розрахунку є графік біоритмів (рис. 4.1) і таблиця даних (табл. 4.1).

Дата народження: 4 вересня 1998 року.

Задана дата розрахунку біоритмів – 15 грудня 2021 року.

Кількість повних прожитих років – H = 22 роки.

Кількість високосних років серед повних прожитих років L = 6 років.

Кількість прожитих днів у рік народження T = 119 днів.

Кількість прожитих днів у поточному році до заданої дати R = 348 днів.

Загальна кількість прожитих днів D = 8503 дні.

Кількість повних прожитих фізичних циклів F = 369

Поточний день фізичного циклу FF = 17

Кількість повних прожитих емоційних циклів E = 303

Поточний день емоційного циклу EE = 20

Кількість повних прожитих інтелектуальних циклів I = 257

Поточний день інтелектуального циклу II = 23

Таблиця 4.1 – Розрахунок біоритмів із заданою датою 15.12.2021

Тип біоритму	Кількість повних циклів	Залишок, днів	Фаза біоритму на дату розрахунку	Дата початку наступного циклу
Фізичний	369	6	Несприятлива	22.12.2021
Емоційний	303	8	Несприятлива	24.12.2021
Інтелектуальний	257	7	Несприятлива	23.12.2021

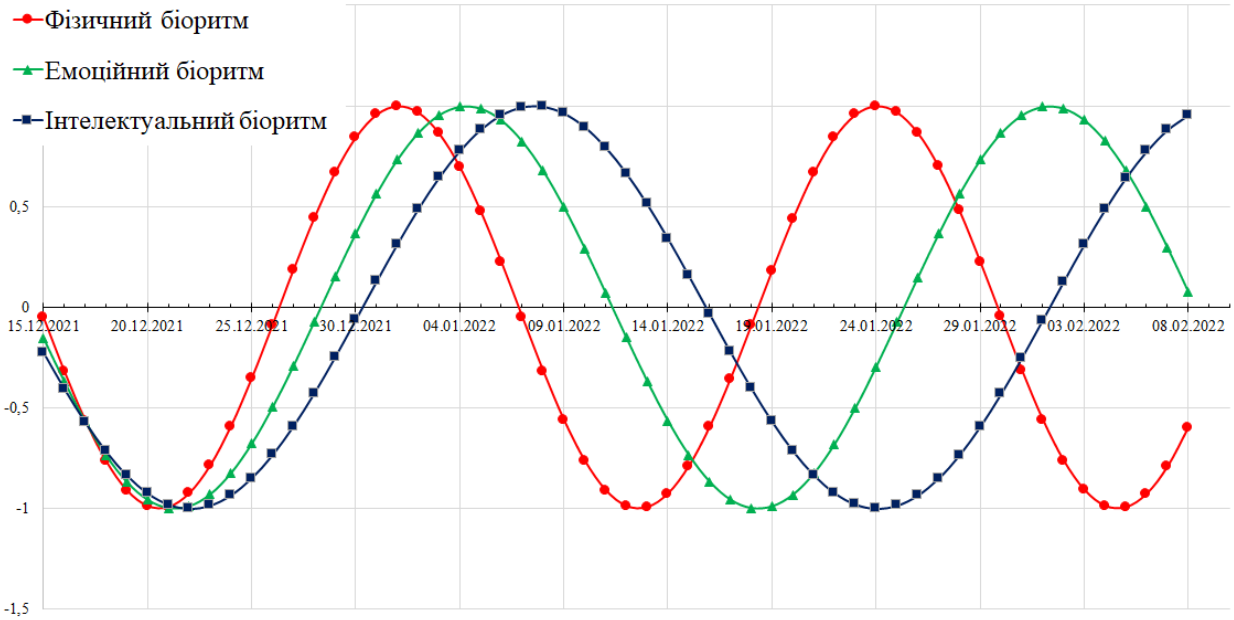

Рисунок 4.1 – Графік персональних біоритмів

За результатом розрахунку персональних біоритмів робимо висновок: найближчий період, в якому одночасно відбуваються зростаюча фізична, емоційна і інтелектуальна активність, можливе досягнення найбільш ефективних результатів триває з 24 грудня 2021 року до 3 січня 2022 року. Несприятливими з погляду фізичного самопочуття будуть дні 12 – 13 січня 2022 року, з погляду емоційної сфери – дні 18 – 19 січня 2022 року, з погляду інтелектуальної сфери –

24 – 25 січня 2022 року. Особливо несприятливими будуть дні 20 – 21 грудня 2021 року, оскільки в ці дні спостерігається закінчення циклів всіх трьох біоритмів. У цей проміжок часу слід бути особливо уважним і, якщо це можливо, уникати стресів, емоційних напружень, навантаження уваги, виснажливої фізичної та розумової праці.

ВИСНОВКИ

В результаті кваліфікаційної роботи виконано розробку та реалізацію автоматизованої веб-системи тестування знань, рівень інформаційної безпеки якої, забезпечує захист інформації від найбільш критичних ризиків, встановлених у відповідності до OWASP Топ-10 2021.

Для досягнення поставленої мети розв’язані наступні задачі.

Виконано огляд педагогічного напрямку дослідження форми проведення та оцінки якості знань у форматі автоматизованого тестування. Представлено особливості проведення тестувань знань, де перелічено проблеми проводження тестування знань в університеті та поставлено цілі на розробку програмного додатку, що буде мати переваги, бути безпечним, ефективним та впровадженим в мережі Інтернет. Розглянуто дистанційне навчання та застосування веб-технологій, що дозволяє студентам навчатися дома, оцінювати освоєний новий матеріал нестандартним способом, але головне – цікавим. Представлені аспекти та переваги електронного оцінювання дозволяють економити час як викладачів, так і студентів; економити витрати на канцелярські товари; впроваджувати нові стратегії навчання та покращувати навчальний процес. Вивчена аналітика дистанційного навчання наголошує, що онлайн-навчання є викликом як для студентів, так і для викладачів, тому що необхідно розуміти в якій формі створювати оцінювальні заходи, щоб вони були максимально ефективними та зрозумілими. Досліджені альтернативні працюючі системи мають як свої переваги, так і недоліки. Впроваджена система буде корисною в першу чергу тим, що всі дані, що зберігаються в базі даних залишаються конференційними, а доступ до них мають лише уповноважені особи.

Розглянуто теоретичні основи захисту веб-додатків. Безпеці веб-додатків необхідно приділяти дуже багато уваги. Безпека веб-додатків залежить від якості програмного коду, від кваліфікації системного адміністратора та від компетенцій усіх користувачів, які мають доступ до чутливої інформації. Вразливості OWASP є дуже шкідливими і в цьому списку зосереджені найнебезпечніші вразливості,

які можуть коштувати купу грошей, або підрив ділової репутації, чи довести аж до втрати бізнесу. Методи самозахисту веб-додатку під час виконання мають деякі складнощі з виявленням вразливостей без штучного налаштування, але з використанням машинного навчання можуть бути корисними у нагоді. Техніки виявлення атак з використанням мови структурованих запитів є необхідними, однак фільтрація даних здається дуже ефективною та найпростішою технікою. Незважаючи на те, що виявлено, що методи виявлення SQLIA, які використовують перевірку введення, схильні до великої кількості хибно-позитивних результатів, і все ж немає 100% гарантії, що немає помилкових негативів.

Представлена практична реалізація захищеної автоматизованої веб-системи тестувань знань. Детально описано вибір засобів реалізації програмного продукту: особливості мов програмування JavaScript та PHP, фреймворків Nuxt та Laravel, інтегрованого середовища розробки PHPStorm. Розглянута архітектура програмного продукту «Тестування знань», яка включає в себе наступні модулі: модуль автентифікації та реєстрації; модуль публікацій; модуль груп; модуль тестування знань; модуль управління користувачами; модуль управління особистою інформацією. Представлені дві технології, що допомогли в вирішенні проблем, які виникли при розробці програмного продукту «Тестування знань» при взаємодії між клієнтською та серверною частинами. Наведені основні складові програмного продукту та інструкція з експлуатації.

ПЕРЕЛІК ПОСИЛАНЬ

1. Mimirinis M. Qualitative differences in academics' conceptions of e-assessment. *Sci-hub*. URL: https://sci-hub.se / https: // www.tandfonline.com / doi / abs / 10.1080 / 02602938.2018.1493087
2. Li X. An examination of a gamified E-quiz system in fostering students' reading habit, interest and ability. *Sci-hub*. URL: https://sci-hub.se/10.1002/pra2.2018.14505501032
3. Alruwais N., Wills G., Wald M. Advantages and Challenges of Using e-Assessment. *International Journal of Information and Education Technology*. 2018. Vol. 8, No. 1. P. 34-37.
4. Kiryakova G. E-assessment-beyond the traditional assessment in digital environment. *International Conference on Technics, Technologies and Education 2020 (ICTTE 2020)*, 4-6 November 2020 Yambol, Bulgaria. P. 1-8.
5. Yin H. Taking e-Assessment Quizzes - A Case Study with an SVD Based Recommender System. *Sci-hub*. URL: https://sci-hub.se/10.1007/978-3-030-03493-1
6. Mouri K. An automatic quiz generation system utilizing digital textbook logs. *Sci-hub*. URL: https: // sci-hub.se / https: // www.tandfonline.com / doi / abs / 10.1080 / 10494820.2019.1620291
7. Petrova T., Ivanova M., Naydenova I. Evaluation of e-assessment: the students' perspective. *The 16th International Scientific Conference eLearning and Software for Education Bucharest*, 23-24 April 2020, Bucharest. P. 199-206.
8. Pevnev V., Popovichenko O., Tsokota Ya. Web application protection technologies. *Сучасні інформаційні системи*. 2020. Т.4, № 1. С. 119-123.
9. Demertzis K. Cognitive Web Application Firewall to Critical Infrastructures Protection from Phishing Attacks. *Journal of Computations & Modelling*. 2012. V.9. P. 1-26.

10. Bach-Nutman M. Understanding The Top 10 OWASP Vulnerabilities. *arXiv*. URL: https://arxiv.org/ftp/arxiv/papers/2012/2012.09960.pdf
11. Zhongxu Y., Zhufeng L., Yan C. A Web Application Runtime Application Self-protection Scheme against Script Injection Attacks. *4th International Conference, ICCCS 2018*, 8-10 June 2018, Haikou, China. P. 566-577.
12. Rua M.T., Musab A.M., Farooq B.A. The impact of sql injection attacks on the security of databases. *6th International Conference on Computing and Informatics, ICOCI 2017*, 25-27 April, 2017, Korea. P. 323-331.
13. JavaScript. *MDN Web Docs*. URL: https: // developer.mozilla.org / ru / docs / Learn / JavaScript
14. Никсон Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5. СПб.: Питер, 2016. 768 с.
15. Макфарланд Д. JavaScript и jQuery: исчерпывающее руководство. Москва: Эксмо, 2015. 880 с.
16. Фрэнаган Д. JavaScript. Подробное руководство. СПб.: Символ-Плюс, 2012. 1080 с.
17. Стефанов С. JavaScript. Шаблоны. СПб.: Символ-Плюс, 2011. 272 с.
18. Гудман Д. JavaScript. Библия пользователя. М.: ООО «И.Д. Вильямс», 2006. 1184 с.
19. Учебник по PHP. *Htmlacademy*. URL: https://htmlacademy.ru/tutorial/php
20. Advance PHP Tutorial. *Phptpoint*. URL: https: // www.phptpoint.com / advanced-php-tutorial/
21. PHP Tutorial. *Tutorialspoint*. URL: https: // www.tutorialspoint.com / php / index.htm
22. PHP Manual. *PHP*. URL: https://www.php.net/manual/en/index.php
23. PHP Tutorial. *w3schools*. URL: https://www.w3schools.com/php/
24. PHP Tutorial - Learn PHP. *Tizag*. URL: http://tizag.com/phpT/
25. The Intuitive Vue Framework. *Nuxtjs*. URL: https://nuxtjs.org/
26. Get started with Nuxt.js on Windows. *Microsoft*. URL: https: // docs.microsoft.com/en-us/windows/dev-environment/javascript/nuxtjs-on-wsl

27. Nuxt.js: Фреймворк для фреймворка Vue.js. *Хабрахабр*. URL: https: // habr.com/ru/post/336902/
28. Load Data from URL Params in Vue.js and Nuxt.js. *Egghead*. URL: https: // egghead.io/lessons/vue-load-data-from-url-params-in-vue-js-and-nuxt-js
29. Preload Data using Promises with Vue.js and Nuxt.js. *Egghead*. URL: https: // egghead.io/lessons/vue-preload-data-using-promises-with-vue-js-and-nuxt-js
30. Add CSS Libraries to Nuxt. *Egghead*. URL: https://egghead.io/lessons/vue-js-add-css-libraries-to-nuxt
31. The PHP Framework for Web Artisans. *Laravel*. URL: https://laravel.com/
32. Laravel – php-фреймворк нового поколения. *Laravel*. URL: https://laravel.su/
33. Laravel Tutorial. *Tutorialspoint*. URL: https: // www.tutorialspoint.com / laravel/index.htm
34. Laravel Tutorial. *w3schools*. URL: https://www.w3schools.in/laravel-tutorial/
35. Laravel framework Tutorial. *Studentstutorial*. URL: https: // www.studentstutorial.com/laravel/laravel-tutorial
36. Laravel localization. *Lokalise*. URL: https://lokalise.com/blog/laravel-localization-step-by-step/
37. HTML5. *Htmlbook*. URL: http://htmlbook.ru/html5
38. HTML HTML5. *Html5book*. URL: https://html5book.ru/html-html5/
39. HTML: HyperText Markup Language. *DevDocs*. URL: https: // devdocs.io/html/
40. HTML Tutorial. *w3schools*. URL: https: // www.w3schools.com / html / default.asp
41. Дронов В.А. HTML5, CSS3 и Web2.0. Разработка современных Web-сайтов. СПб.: БХВ-Петербург, 2011. 416 с.
42. Хоган Б. HTML5 и CSS3. Веб-разработка по стандартам нового поколения. СПб.: Питер, 2014. 320 с.
43. CSS CSS3. *Html5book*. URL: https://html5book.ru/css-css3/
44. CSS. *Htmlbook*. URL: http://htmlbook.ru/samcss

45. CSS. *DevDocs*. URL: https://devdocs.io/css/
46. CSS Tutorial. *w3schools*. URL: https://www.w3schools.com/css/default.asp
47. Пьюривал С. Основы разработки веб-приложений. СПб.: «Питер», 2015. 272 с.
48. Мейер Э., Уэйл Э. CSS: полный справочник. СПб.: «Диалектика», 2019. 1088 с.
49. ГОСТ 12.0.003-74. ССБП. Небезпечні і шкідливі виробничі фактори. Класифікація.
50. ГОСТ 12.1.006 – 84 ССБП. Електромагнітні поля радіочастот. Допустимі рівні на робочих місцях та вимоги до проведення контролю.
51. Шуаібов О.К. Практикум з охорони праці. Навчальний посібник. Ужгород: Ужгородський національний університет, фізичний факультет, 2007. 280 с.
52. Сафонов В.В. Інженерні рішення з охорони праці при розробці дипломних проектів інженерно-будівельних спеціальностей: Навчальний посібник. К.: Основа, 2011. 480 с.