

Національний лісотехнічний університет України
(повне найменування вищого навчального закладу)

Навчально-науковий інститут деревообробних та
комп'ютерних технологій і дизайну
(повне найменування інституту, назва факультету (відділення))

Кафедра інформаційних технологій
(повна назва кафедри (предметної, циклової комісії))

Пояснювальна записка

до дипломної роботи

другий (магістерський)

(рівень вищої освіти)

на тему: Розроблення інформаційної системи “Сервісне обслуговування
техніки кафедри ІТ НЛТУ України”

Виконав: студент б курсу групи КН-61м
спеціальності

122 “Комп’ютерні науки”

(шифр і назва напрямку підготовки, спеціальності)

Смірнов Є.В.

(прізвище та ініціали)

Керівник Сторожук О.Л.

(прізвище та ініціали)

Рецензент _____

(прізвище та ініціали)

Національний лісотехнічний університет України
(повне найменування вищого навчального закладу)

ННІ деревообробних та комп'ютерних технологій і дизайну

Кафедра інформаційних технологій

Рівень вищої освіти другий (магістерський)

Спеціальність 122 “Комп'ютерні науки”

(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Крошній І.М.

“ ” _____ 20__ року

З А В Д А Н Н Я
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Смірнову Євгенові Володимировичу

(прізвище, ім'я, по батькові)

1. Тема роботи Розроблення інформаційної системи “Сервісне обслуговування техніки кафедри ІТ НЛТУ України”

керівник роботи Сторожук Олександр Леонідович, к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від “20”12 2021 року № С-617

2. Термін подання студентом роботи 12.12.2022р.

3. Вихідні дані до роботи: Аналіз стану проблемної області. Визначення інформаційного та математичного забезпечення. Обґрунтування вибору обраних засобів розробки інформаційної системи.

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити): 1) Стан проблемної області; 2) Інформаційне забезпечення; 3) Математичне забезпечення; 4) Програмне забезпечення; 5) Розроблення стартап проекту.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Презентація до диплому

6. Дата видачі завдання 20 грудня 2021 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1.	<i>Огляд літературних даних.</i>	<i>20.12.2021р. – 13.02.2022р.</i>	
2.	<i>Розділ 1. Стан проблемної області.</i>	<i>14.02.2022р. – 06.03.2022р.</i>	
3.	<i>Розділ 2. Інформаційне забезпечення.</i>	<i>07.03.2022р. – 24.04.2022р.</i>	
4.	<i>Розділ 3. Математичне забезпечення.</i>	<i>25.04.2022р. – 22.05.2022р.</i>	
5.	<i>Розділ 4. Програмне забезпечення.</i>	<i>23.05.2022р. – 28.08.2022р.</i>	
6.	<i>Розділ 4. Розроблення стартап проекту .</i>	<i>29.08.2022р. – 25.09.2022р.</i>	
7.	<i>Аналіз отриманих результатів та написання висновків. Оформлення дипломної роботи.</i>	<i>26.09.2022р. – 11.12.2022р.</i>	
8.	<i>Здача пояснювальної записки на перевірку та здача роботи рецензенту.</i>	<i>12.12.2022 р.</i>	

Студент

(підпис)

Смірнов Є.В.

(прізвище та ініціали)

Керівник роботи

(підпис)

Сторожук О.Л.

(прізвище та ініціали)

РЕФЕРАТ

Дипломна робота містить 73 сторінки пояснювальної записки, 20 рисунків, 18 таблиць, 1 додаток, 9 джерел.

В даній роботі розроблено інформаційну систему “Сервісне обслуговування техніки кафедри ІТ НЛТУ України”. Реалізовано динамічне створення QR-коду із закодованим URL-посиланням на сервісну сторінку підтримки одиниці техніки. Для даної сторінки реалізовано тест CAPTCHA, щоб позбавитись спаму від інтернет-ботів. Адміністратор інформаційної системи бачить усі запити щодо виявлених несправностей та відповідальних за їх усунення. Розроблений підхід забезпечить своєчасне вивлення поломок та пришвидшить ремонт техніки.

Ключові слова: *Ajax, Jquery, AngularJS, HTML, CSS, JS, PhP, MySQL.*

ABSTRACT

The thesis contains 73 pages of explanatory note, 20 figures, 18 tables, 1 appendices, 9 sources.

Thesis is devoted to the development information system "Service of equipment of the IT department of UNFU". The dynamic implementation of QR-code with coded URL-link to the service page of the unit support. A CAPTCHA test has been implemented for this page in order to get rid of spam from Internet bots. The administrator of the information system sees all requests regarding detected malfunctions and those responsible for their elimination. The developed approach will ensure timely detection of breakdowns and speed up the repair of equipment.

Keywords: *Ajax, Jquery, AngularJS, HTML, CSS, JS, PhP, MySQL.*

ТЕХНІЧНЕ ЗАВДАННЯ

Розробити інформаційну систему “Сервісне обслуговування техніки кафедри ІТ НЛТУ України” за допомогою таких засобів: AngularJS, HTML, CSS, JS, PHP, MySQL.

Реалізувати динамічне створення QR-коду із закодованим URL-посиланням на персоналізовану сервісну сторінку підтримки одиниці техніки. На даній сторінці викладач, інженер, студент має мати можливість залишити відгук про виявлену поломку і відправити його відповідальному за ремонт техніки. Передбачити можливість зворотнього зв'язку, щоб повідомити про усунення поломки. Для даної сторінки реалізувати тест CAPTCHA, який має захищати розроблену інформаційну систему від спаму інтернет-ботів.

Адміністратор інформаційної системи має бачити усі запити щодо виявлених несправностей та відповідальних за їх усунення.

ЗМІСТ

ВСТУП	6
РОЗДІЛ 1. СТАН ПРОБЛЕМНОЇ ОБЛАСТІ	7
1.1. Огляд проблемної області	7
1.2. Програми конкуренти.	8
РОЗДІЛ 2. ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ	9
2.1. Технологія AJAX	9
2.2. Бібліотека JQUERY	9
2.3. Фреймворк ANGULAR JS	10
2.4. Мова HTML	11
2.5. Мова стилю сторінок CSS	13
2.6. Мова JS	14
2.7. Мова PHP	15
2.8. Система керування реляційними базами даних MYSQL	16
РОЗДІЛ 3. МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ	18
3.1. Алгоритм генерації QR-коду	18
РОЗДІЛ 4. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ	24
4.1. Visual Studio Code	24
4.2. Програма MAMP	28
4.3. Послідовний опис розробки веб орієнтованої інформаційної системи ...	31
РОЗДІЛ 5. РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ	44
5.1. Опис ідеї проекту	44
5.2. Аналіз технологічних можливостей реалізації ідей проекту	46
5.3. Аналіз ринкових можливостей запуску стартап-проекту	46
5.4. Розроблення ринкової стратегії проекту	52
5.5. Розроблення маркетингової програми стартап-проекту	53
ВИСНОВКИ	57
ДОДАТКИ	59

ВСТУП

Сучасне життя важко уявити без техніки яка нас оточує. В сучасних умовах для навчального процесу в університетах задіюється велика кількість комп'ютерної та іншої техніки. Інженерам доводиться обслуговувати її і підтримувати у робочому стані.

Актуальність проблеми.

Для навчального процесу важливо, щоб уся техніка була робочою, а у випадку її виходу із ладу була відремонтована у найшвидші терміни. Поломки бувають як фізичного плану так і програмного. Для прикладу деколи після оновлення програмного забезпечення починають працювати некоректно інші програми. Для швидкого вирішення цієї проблеми необхідно у стислі терміни повідомити відповідального інженера. Різні аудиторії закріплені за окремими працівниками і не завжди користувач знає до кого звертатися.

Для пришвидшення та покращення сервісного обслуговування техніки кафедри ІТ НЛТУ України було вирішено розробити інформаційну систему яка має вирішити описанні проблеми.

Об'єктом дослідження є використання фреймворку AngularJS під час розробки інформаційної системи.

Предмет дослідження – практичне використання Ajax, JQuery, AngularJS, HTML, CSS, JS, PHP, MySQL для розробки інформаційної системи.

Метою роботи є розробка інформаційної системи “Сервісне обслуговування техніки кафедри ІТ НЛТУ України”.

Практичне значення отриманих результатів. Розроблена інформаційна система сервісного обслуговування техніки кафедри ІТ НЛТУ України забезпечує своєчасне вивлення поломок техніки, та зменшує терміни які необхідні для їх усунення.

РОЗДІЛ 1. СТАН ПРОБЛЕМНОЇ ОБЛАСТІ

1.1. Огляд проблемної області

Форма зворотного зв'язку - це спосіб зібрати відгуки про послуги компанії. Мета полягає в тому, щоб краще зрозуміти загальний досвід споживачів, щоб визначити напрямки вдосконалення.

Є багато різних способів отримати відгук від клієнтів. Приклади форм зворотного зв'язку можуть включати:

- спливаючі вікна на веб-сайті;
- чат-боти;
- електронні листи.

Вони також можуть варіюватися в межах від одного питання до більш поглибленого опитування. Більшість компаній використовують декілька основних типів шаблонів зворотного зв'язку, але існує багато різновидів подання цих форм. Запити на зворотний зв'язок можуть відображатися як у спливаючих вікнах на веб-сторінках, чат-ботах або електронних листах. Тип обраної форми зворотного зв'язку із клієнтами буде залежати від того, яку інформацію необхідно отримати.

Також перевагою є те, що користувач, використовуючи форму зворотного зв'язку, не розкриває свою e-mail адресу спамерам. Саме використання контактної форми має покращити комунікацію між викладачами та інженерним персоналом на кафедрі інформаційних технологій НЛТУ України. Адже набагато простіше відсканувати qr-код та заповнити не велику форму і відповідальний персонал вже буде повідомлений про ту чи іншу проблему. Також після завершення ремонтних робіт, адміністратор може надіслати користувачеві який надіслав заявку листа з повідомленням про завершення своєї роботи.

Отож основні плюси використання такої системи:

- економія часу викладача;
- майже миттєвий відгук від інженера;
- не потрібна безпосередня присутність викладача для роз'яснення проблеми;
- простота використання.

1.2. Програми конкуренти.

Програм конкурентів нашому веб-застосунку не було знайдено. Проте існує велика кількість ІТ компаній які використовують схожі веб-орієнтовані сервіси у які вмонтовано форми зворотного зв'язку для працівників компанії. Унікальність цієї роботи полягає у використанні QR-code, розміщеного на блоці техніки, або поруч, для переходу на персоналізовану сервісну сторінку підтримки одиниці техніки. Дана сторінка із даними про техніку містить поле для можливості залишити відгук про виявлену поломку і відправити його відповідальному за ремонт техніки. Також сторінка захищена від спаму інтернет-ботів за допомогою тесту CAPTCHA.

Висновки до розділу

У даному розділі проаналізовано проблемну область та описано основні аспекти які повина вирішувати майбутня інформаційна система.

Також обґрунтовано застосування QR-code, який буде розміщуватися на блоці техніки, або поруч. Даний QR-cod забезпечить швидкий перехід на персоналізовану сервісну сторінку підтримки одиниці техніки.

РОЗДІЛ 2. ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ

2.1. Технологія AJAX

AJAX = Асинхронний JavaScript та XML. AJAX не є мовою програмування. AJAX використовує комбінацію:

- вбудованих у браузері об'єкти XMLHttpRequest (для запиту даних з веб-сервера);
- JavaScript і HTML DOM (для відображення або використання даних).

Програми AJAX можуть використовувати XML для транспортування даних, але також можуть передавати дані як звичайний текст або JSON-текст.

AJAX дозволяє асинхронно оновлювати веб-сторінки шляхом обміну даними із веб-сервером. Це означає, що можна оновлювати частини веб-сторінки, не перезавантажуючи цілу сторінку.

Термін Ajax позначає широкую групу веб-технологій, які можна використовувати для реалізації веб-додатків, які взаємодіють із сервером у фоновому режимі, не втручаючись у поточний стан сторінки:

- HTML (або XHTML) та CSS для презентації;
- об'єктна модель документа (DOM) для динамічного відображення та взаємодії з даними;
- JSON або XML для обміну даними та XSLT для обробки XML;
- об'єкт XMLHttpRequest для асинхронного зв'язку;
- JavaScript для об'єднання цих технологій. [1].

2.2. Бібліотека JQUERY

jQuery - це бібліотека JavaScript "писати менше, робити більше".

jQuery значно спрощує на веб-сайті використання JavaScript.

Якщо для виконання завдань необхідно використовувати багато рядків коду JavaScript то варто застосовувати jQuery який обертає їх у методи, які можна викликати одним рядком коду.

Також для спрощення складних речей з JavaScript, зокрема викликів AJAX та маніпуляції DOM варто застосовувати jQuery.

Бібліотека jQuery містить функції які забезпечують роботу з:

- HTML / DOM;
- CSS;
- методами подій HTML;
- ефектами та анімацією.

Існує безліч інших бібліотек JavaScript, але jQuery, мабуть, найпопулярніший, а також найбільш поширений.

Багато великих компаній, зокрема IBM, Google, Netflix, Microsoft використовують jQuery. [1].

2.3. Фреймворк ANGULAR JS

AngularJS - це структурний фреймворк для динамічних веб-програм. Це дозволяє використовувати HTML як мову вашого шаблону і дозволяє розширити синтаксис HTML, щоб чітко і коротко виразити компоненти додатка. Прив'язка даних AngularJS та ін'єкція залежностей усувають значну частину коду, який інакше довелося б написати. І все це відбувається в браузері, що робить його ідеальним партнером для будь-якої серверної технології.

AngularJS - це те, яким би був HTML, якби він був розроблений для додатків. HTML - чудова декларативна мова для статичних документів.

Невідповідність імпедансу між динамічними додатками та статичними документами часто вирішується за допомогою:

- бібліотек - набір функцій, корисних при написанні веб-програм, наприклад, jQuery;

- фреймворку - інфраструктура програмних рішень, що полегшує розробку складних систем.

AngularJS використовує інший підхід. Він намагається мінімізувати невідповідність імперативу між орієнтованим на документ HTML і тим, що потрібно додатку, створюючи нові конструкції HTML. AngularJS вчить браузер новому синтаксису через конструкцію, яку ми називаємо директивами.

Директиви включають:

- прив'язку даних, як у `{{}}`;
- структури управління DOM для повторення, показу та приховування фрагментів DOM;
- підтримку форм та перевірки форм;
- додавання нової поведінки до елементів DOM, таких як обробка подій DOM;
- групування HTML у багаторазові компоненти.

AngularJS - це не єдиний фрагмент загальної головоломки побудови веб-додатків на стороні клієнта. Він обробляє весь код DOM і AJAX, який ви колись писали вручну, і вкладає його в чітко визначену структуру. Це змушує AngularJS відповідати, як слід будувати додаток CRUD (створити, читати, оновити, видалити). Все, що потрібно для створення CRUD-додатків у згуртованому наборі: прив'язка даних, основні директиви щодо шаблонів, перевірка форми, маршрутизація, глибоке зв'язування, багаторазові компоненти та залежності [6].

2.4. Мова HTML

Hypertext Markup Language (HTML) - це комп'ютерна мова, яка складає більшість веб-сторінок та онлайн-додатків. Гіпертекст - це текст, який використовується для посилання на інші фрагменти тексту, тоді як мова розмітки - це ряд маркувань, який повідомляє веб-серверам стиль і структуру документа.

HTML не вважається мовою програмування, оскільки він не може створити динамічну функціональність. Натомість за допомогою HTML веб-користувачі можуть створювати та структурувати розділи, абзаци та посилання за допомогою елементів, тегів та атрибутів.

Ось деякі найпоширеніші способи використання HTML:

- *Веб-розробка.* Розробники використовують HTML-код для розробки елементів веб-сторінки, такі як текст, гіперпосилання та медіа-файли;
- *Інтернет-навігація.* Користувачі можуть легко переходити та вставляти посилання між відповідними сторінками та веб-сайтами, оскільки HTML інтенсивно використовується для вбудовування гіперпосилань;
- *Веб-документація.* HTML дозволяє впорядковувати та форматовувати документи, подібно до Microsoft Word.

Також варто зазначити, що HTML зараз вважається офіційним веб-стандартом. Консорціум Всесвітньої павутини (W3C) підтримує та розробляє специфікації HTML, а також забезпечує регулярне оновлення.

Середній веб-сайт включає кілька різних HTML-сторінок. Наприклад, домашня сторінка, сторінка про інформацію та сторінка контактів, які матимуть окремі файли HTML.

Документи HTML - це файли, які закінчуються розширенням .html або .htm. Веб-браузер читає файл HTML і відтворює його вміст, щоб користувачі Інтернету могли його переглянути.

Усі HTML-сторінки мають ряд HTML-елементів, що складаються з набору атрибутів та тегів. Будівельними блоками веб-сторінки є елементи HTML. Для повідомлення веб-браузеру, де елемент починається та закінчується використовують теги. У свою чергу атрибути описують характеристики елемента [2].

2.5. Мова стилю сторінок CSS

CSS означає «Каскадні таблиці стилів». І саме у CSS робиться наголос на «Стиль». HTML використовується для структурування веб-документа CSS визначає стиль документа - макети сторінок, кольори та шрифти визначаються за допомогою CSS. Подумайте про HTML, як про основу, а CSS як про естетичний вибір

CSS забезпечує стиль у веб-сторінки, взаємодіючи з елементами HTML.

Каскадні таблиці стилів, які називають CSS, - це проста мова дизайну, призначена для спрощення процесу верстки веб-сторінки, щоб вона стала презентабельною.

CSS обробляє зовнішній вигляд частини веб-сторінки. За допомогою CSS можна керувати кольором тексту, стилем шрифтів, інтервалом між абзацами, розміром та розміщенням стовпців, фоновими зображеннями або кольорами, що використовуються, варіаціями відображення, дизайном макета для різних пристроїв та розмірами екрану, а також безліччю інших ефектів.

Найчастіше CSS поєднується з мовами розмітки HTML або XHTML. CSS легко вивчити і зрозуміти, також він забезпечує потужний контроль над поданням HTML-документа.

Переваги CSS

CSS під час оформлення HTML сторінок економить час, оскільки достатньо написати CSS один раз, а потім повторно на декількох HTML-сторінках використовувати один і той же стиль. Отже можна застосувати стиль який визначили для елементів HTML до великої кількості веб-сторінок.

Якщо використовувати CSS сторінки завантажуються швидше оскільки не потрібно кожного разу писати атрибути тегів HTML. Це досягається завдяки тому, що для тегу достатньо написати одне правило CSS та застосувати його до всіх випадків появи цього тегу. Пришвидшуємо час завантаження завдяки зменшенню коду.

Простота обслуговування

Для внесення глобальних змін достатньо змінити стиль, і всі елементи які розташовані на всіх веб-сторінках будуть оновлені автоматично.

Покращені стилі HTML

У CSS є ширший набір атрибутів, ніж у HTML.

Сумісність декількох пристроїв

Таблиці стилів дозволяють оптимізувати вміст для більш ніж одного типу пристроїв [3].

2.6. Мова JS

JavaScript (часто скорочується до JS) - об'єктно-орієнтована мова яка містить сучасні функції та є найвідомішою мовою сценаріїв веб-сторінок, але вона також використовується і в багатьох середовищах, які не належать до браузера.

Це мова сценаріїв, що базується на прототипі, з використанням багатьох парадигм, яка є динамічною та підтримує об'єктно-орієнтовані, імперативні та функціональні стилі програмування.

Оскільки JavaScript працює на клієнтській стороні Інтернету, її досить часто використовують для проектування/програмування поведінки веб-сторінок у випадку виникнення події. JavaScript - потужна мова сценаріїв, яка широко застосовується для контролю поведінки веб-сторінок.

JavaScript може функціонувати як процедурна так і об'єктно-орієнтована мова. Об'єкти створюються програмно в JavaScript, приєднуючи методи та властивості до порожніх об'єктів під час виконання, на відміну від синтаксичних класів, поширених у компільованих мовах, таких як C++ та Java. Після побудови об'єкта його можна використовувати як план (або прототип) для створення подібних об'єктів.

Динамічні можливості JavaScript включають побудову об'єкта часу виконання, списки параметрів змінних, функціональні змінні, створення динамічного скрипта (через eval), інтроспекцію об'єкта (через for ... in) та відновлення вихідного коду (програми JavaScript можуть декомпілювати тіла функцій назад у вихідний текст).

Безумовно, найпоширенішим середовищем хостів для JavaScript є веб-браузери. Веб-браузери зазвичай використовують загальнодоступний API для створення хост-об'єктів, відповідальних за відображення DOM в JavaScript.

Іншим поширеним додатком для JavaScript є мова сценаріїв на веб-сервері. Веб-сервер JavaScript буде виставляти об'єкти хосту, що представляють HTTP-запити та об'єкти відповіді, якими потім може маніпулювати програма JavaScript для динамічного створення веб-сторінок. Node.js - популярний приклад цього.

Поширені приклади JavaScript - вікно пошуку на Amazon, відеозапис новин, вбудований у The New York Times, або оновлення вашої стрічки Twitter [4].

2.7. Мова PHP

PHP - це мова сценаріїв на стороні сервера, що використовується для розробки статичних веб-сайтів або динамічних веб-сайтів чи веб-додатків. PHP розшифровується як попередній процесор гіпертексту, що раніше означало особисті домашні сторінки.

Сценарії PHP можна інтерпретувати лише на сервері, на якому встановлений PHP. Клієнтські комп'ютери, які отримують доступ до скриптів PHP, потребують лише веб-браузера.

PHP-файл містить теги PHP і закінчується розширенням ".php".

Сценарій - це набір інструкцій з програмування, який інтерпретується під час виконання.

Серверні сценарії інтерпретуються на сервері, тоді як сценарії на стороні клієнта інтерпретуються клієнтською програмою.

Призначення сценаріїв, як правило, полягає у підвищенні продуктивності або виконанні рутинних завдань для програми.

PHP - це скрипт на стороні сервера, який інтерпретується на сервері, тоді як JavaScript - приклад сценарію на стороні клієнта, який інтерпретується клієнтським браузером. Отже можна стверджувати, що і PHP, і JavaScript можуть бути вбудовані в HTML-сторінки [5].

2.8. Система керування реляційними базами даних MySQL

Шведська компанія під назвою MySQL AB спочатку розробила MySQL у 1994 році. Потім американська технологічна компанія Sun Microsystems придбала MySQL AB у 2008 році. Технологічний гігант Oracle у 2010 році придбав Sun Microsystems, і MySQL практично належить Oracle.

Що стосується загального визначення, MySQL - це система керування реляційними базами даних з відкритим кодом (СУБД) із моделлю клієнт-сервер. СУБД - це програмне забезпечення чи послуга, що використовується для створення та управління базами даних на основі реляційної моделі.

Основні процеси, що відбуваються в середовищі MySQL, однакові, а саме:

- MySQL створює базу даних для зберігання та обробки даних, визначаючи взаємозв'язок кожної таблиці;
- клієнти можуть робити запити, вводячи конкретні оператори SQL на MySQL;
- серверна програма відповідає на інформацію, що запитується, і вона з'явиться на стороні клієнтів.

З боку клієнтів - вони зазвичай вибирають, який графічний інтерфейс MySQL використовувати. Чим легший і зручніший графічний інтерфейс, тим швидше і простіше буде їх діяльність з управління даними. Одними з

найпопулярніших графічних інтерфейсів MySQL є MySQL WorkBench, SequelPro, DBVisualizer та Navicat DB Admin Tool. Деякі з них безкоштовні, тоді як інші є комерційними, деякі працюють виключно для macOS, а інші сумісні з основними операційними системами. Клієнти повинні обирати графічний інтерфейс залежно від своїх потреб. Для управління веб-базами даних, включаючи сайт WordPress, найбільш очевидним напрямком є phpMyAdmin [7].

Висновки до розділу

У даному розділі було охарактеризовано обрані мови програмування, використаний фреймворк AngularJS та інші технології на основі яких буде розроблена інформаційна система “Сервісне обслуговування техніки кафедри ІТ НЛТУ України”.

РОЗДІЛ 3. МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ

3.1. Алгоритм генерації QR-коду

QR код – це монохромна картинка, на якій деякі пристрої (наприклад, смартфон зі спеціальним додатком) розпізнають текст. Цим текстом може бути не лише проста фраза, а для прикладу, хоч це й не входить до офіційної специфікації, URL - посилання, номера телефону чи візитної картки. Такі коди найчастіше використовують, щоб закодувати посилання та роздрукувати його на плакаті чи візитці.

Процес генерації QR коду поділяється на декілька точних кроків:

- кодування даних;
- додавання службової інформації та заповнення;
- поділ інформації на блоки;
- створення байтів корекції;
- об'єднання блоків;
- розміщення інформації на QR-коді.

Кодування даних

Залежно від того, які символи використовуються для кодування даних, QR код підтримує кілька способів кодування: кодування побайтове, кандзі (китайсько-японські ієрогліфи), буквено-цифрове та цифрове. Байти кодування не вимагають взагалі, спочатку необхідно створити порожню послідовність біт, яку будемо заповнювати. Кандзі розглядати не будемо. Буквено цифрове – цифри та символи \$%*+-./: і пробіл, великі літери латинського алфавіту. Цифрове кодування передбачає використання лише цифр від 0 до 9.

Цифрове кодування

Для даного типу кодування використовують 10 біт на кожні 3 символи. Послідовність символів розбиваємо на групи по 3 цифри. На наступному кроці тризначне число із кожної групи перетворюємо на 10-бітове двійкове число і

тепер додаємо до послідовності біт. Розглянемо випадок якщо загальна кількість символів не кратна 3. У такому разі наступні дії визначаються кількістю символів які залишились. Якщо в кінці залишається 2 символи, отримане двозначне число кодується 7 бітами, а якщо 1 символ, 4 бітами.

Приклад: необхідно закодувати рядок із цифр від одного до восьми. Якщо розіб'ємо даний рядок на числа то отримаємо 123, 456 та 78. Тепер переведемо отримані числа у двійковий вигляд. Для 123 це буде 0001111011, відповідно для 456 – 0111001000 та 78 – 0111001000. Після об'єднання отримуємо потік: 000111101101100

Літерно-цифрове кодування

Для даного кодування використовують одинадцять біт інформації на два символи. Вхідний потік символів ділимо на групи по два символи. Відповідно до таблиці 3.1 кожен символ кодуємо за наступним правилом: на 45 множимо значення першого символу в групі і після цього результат додаємо до другого символу. Отримане число додаємо до послідовності біт. У випадку якщо маємо в останній групі один символ, його значення кодуємо шестибітним числом. Отриманий результат додаємо до послідовності біт.

Таблиця 3.1. Значення символів у буквенно-цифровому кодуванні

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
15	16	17	18	19	20	21	22	23	24	25	26	27	28	29

U	V	W	X	Y	Z	Пробіл	\$	%	*	+	-	.	/	:
30	31	32	33	34	35	36	37	38	39	40	41	42	43	44

Приклад: необхідно закодувати слово «HELLO». Відповідно до вищенаведеного алгоритму, розбиваємо дане слово на групи по два символи HE, LL, O; згідно таблиці 3.1 кодуємо кожен символ 17, 14; 21, 21 та 24. Тепер знаходимо значення для кодної групи окремо. Для першої $17 \cdot 45 + 14 = 779$, для другої $21 \cdot 45 + 21 = 966$ та третьої $24 = 24$. Отримані значення переводимо у двійковий вигляд і отримуємо: 01100001011, 01111000110, 011000. Після об'єднання отримуємо наступну послідовність біт: 0110000101101111000110011000

Побайтове кодування

Дане кодування є універсальним, оскільки дозволяє закодувати будь-які символи. Але воно має і свій недолік – відносно низьку густину інформації. У цьому випадку текст кодується в будь-якому кодуванні (рекомендовано UTF-8) і отримана послідовність байт береться в незмінному вигляді.

Додавання службової інформації

Що вищий рівень корекції, то вищий допустимий рівень пошкодження зображення. Усього є 4 рівні корекції: L (максимум 7% пошкоджень), M (15%), Q (25%) та H (30%). Найчастіше використовується рівень M. Якщо необхідно додати на QR код свій малюнок то необхідно використовувати рівень H.

Ще одна властивість QR коду - його версія (чим вона більша, тим більший розмір). Усього існує 40 версій. Номер версії залежить від кількості інформації, що кодується, і від рівня корекції. У таблиці 3.2 вказано максимальну кількість корисної інформації разом із службовою (у бітах), яку можна закодувати у QR коді певної версії. Рядок – рівень корекції, стовпець – номер версії.

Додавання службових полів

До цього моменту вже має бути обраний рівень корекції та визначено версію. Тепер треба перед послідовністю біт, отриманої в попередньому пункті, додати на початку два поля: спосіб кодування та кількість даних. Спосіб кодування - поле довжиною 4 біта, яке має наступні значення: 0001 для цифрового кодування, 0010 для буквено-цифрового та 0100 для побайтового.

Таблиця 3.2. Максимальна кількість інформації яку можна закодувати у QR кодї

	1	2	3	4	5	6	7	8	9	10
L	152	272	440	640	864	1088	1248	1552	1856	2192
M	128	224	352	512	688	864	992	1232	1456	1728
Q	104	176	272	384	496	608	704	880	1056	1232
H	72	128	208	288	368	480	528	688	800	976

	11	12	13	14	15	16	17	18	19	20
L	2592	2960	3424	3688	4184	4712	5176	5768	6360	6888
M	2032	2320	2672	2920	3320	3624	4056	4504	5016	5352
Q	1440	1648	1952	2088	2360	2600	2936	3176	3560	3880
H	1120	1264	1440	1576	1784	2024	2264	2504	2728	3080

	21	22	23	24	25	26	27	28	29	30
L	7456	8048	8752	9392	10208	10960	11744	12248	13048	13880
M	5712	6256	6880	7312	8000	8496	9024	9544	10136	10984
Q	4096	4544	4912	5312	5744	6032	6464	6968	7288	7880
H	3248	3536	3712	4112	4304	4768	5024	5288	5608	5960

	31	32	33	34	35	36	37	38	39	40
L	14744	15640	16568	17528	18448	19472	20528	21616	22496	23648
M	11640	12328	13048	13800	14496	15312	15936	16816	17728	18672
Q	8264	8920	9368	9848	10288	10832	11408	12016	12656	13328
H	6344	6760	7208	7688	7888	8432	8768	9136	9776	10208

Кількість даних - це кількість символів, що кодуються, а для побайтового - кількість байт (а не біт в отриманій послідовності), представлене у вигляді двійкового числа певної довжини (визначається за таблицею 3.3).

Таблиця 3.3 Довжина поля даних

	Версія 1–9	Версія 10–26	Версія 27–40
Цифрове	10 біт	12 біт	14 біт
Літерно-цифрове	9 біт	11 біт	13 біт
Побайтове	8 біт	16 біт	16 біт

Приклад: задано рядок довжиною 100 байт, закодований побайтово, рівень корекції - М. Довжина послідовності біт цього рядка - 800 біт. Skorиставшись таблицею 3.2 можна визначити, що найоптимальніше буде використовувати шосту версію. Довжина поля 8 біт (таблиця 3.3). Поле, що визначає спосіб кодування має вигляд 0100, поле кількості даних - 01100100 (100 у двійковому вигляді). У результаті вийде послідовність біт 010001100100<початкова послідовність>.

Якщо довжина отриманої послідовності біт виявилася більшою за допустиму для обраної версії, то версію треба збільшити на одну і зробити додавання службових полів заново.

Специфікація припускає використання змішаного кодування. Це означає, що кілька груп даних можна закодувати різними способами та об'єднати в одну послідовність. Це робиться таким чином: <спосіб кодування даних 1><кількість даних 1><дані 1><спосіб кодування даних 2><кількість даних 2><дані 2> і так далі [9].

Висновки до розділу

В даному розділі було розглянуто теоретичні основи на яких базується алгоритм генерації QR-коду. Описано основні етапи та для кращого розуміння наведено приклади, які їх описують. На основі даного алгоритму, реалізовано генерацію QR-коду для кожної одиниці техніки, що введена у розроблену інформаційну систему “Сервісне обслуговування техніки кафедри ІТ НЛТУ України”

РОЗДІЛ 4. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

4.1. Visual Studio Code

Visual Studio Code - це редактор вихідного коду, створений Microsoft для Windows, Linux та macOS. Особливості включають підтримку налагодження, підсвічування синтаксису, інтелектуальне заповнення коду, фрагменти, рефакторинг коду та вбудований Git. Користувачі можуть змінювати тему, комбінації клавіш, налаштування та встановлювати розширення, що додають додаткову функціональність.

Корпорація Майкрософт випустила більшість вихідного коду Visual Studio Code у сховищі microsoft / vscode GitHub, використовуючи ім'я "Код - OSS" під ліцензією MIT, тоді як версії Microsoft є власною безкоштовною програмою.

У опитуванні розробників Stack Overflow 2019 Visual Studio Code було визнано найпопулярнішим інструментом середовища для розробників, 50,7% з 87 317 респондентів повідомили, що вони ним користуються.

Visual Studio Code вперше був оголошений 29 квітня 2015 року корпорацією Майкрософт на конференції 2015 року. Незабаром після цього було випущено збірку Preview.

18 листопада 2015 року Visual Studio Code вийшов під ліцензією MIT, вихідний код якого доступний на GitHub. Також була оголошена підтримка розширення. 14 квітня 2016 р. Visual Studio Code завершив загальнодоступний етап попереднього перегляду та був випущений в Інтернет.

Функції, які Visual Studio Code включає в себе нестандартно, - це лише початок. Розширення, рис.3.1 VS Code дозволяють додавати мови, налагоджувачі та інструменти до інсталяції для підтримки робочого процесу розробки. Багата модель розширюваності VS Code дозволяє авторам розширень підключатись безпосередньо до інтерфейсу VS Code та надавати функціональність за допомогою тих самих API, що використовуються VS Code.

Ви можете переглядати та встановлювати розширення з VS Code. Відкрийте вікно розширень, натиснувши піктограму розширень на панелі активності збоку від коду VS або команду Перегляд: розширення (Ctrl + Shift + X).

Це покаже вам список найпопулярніших розширень VSCode на VS Code Marketplace.

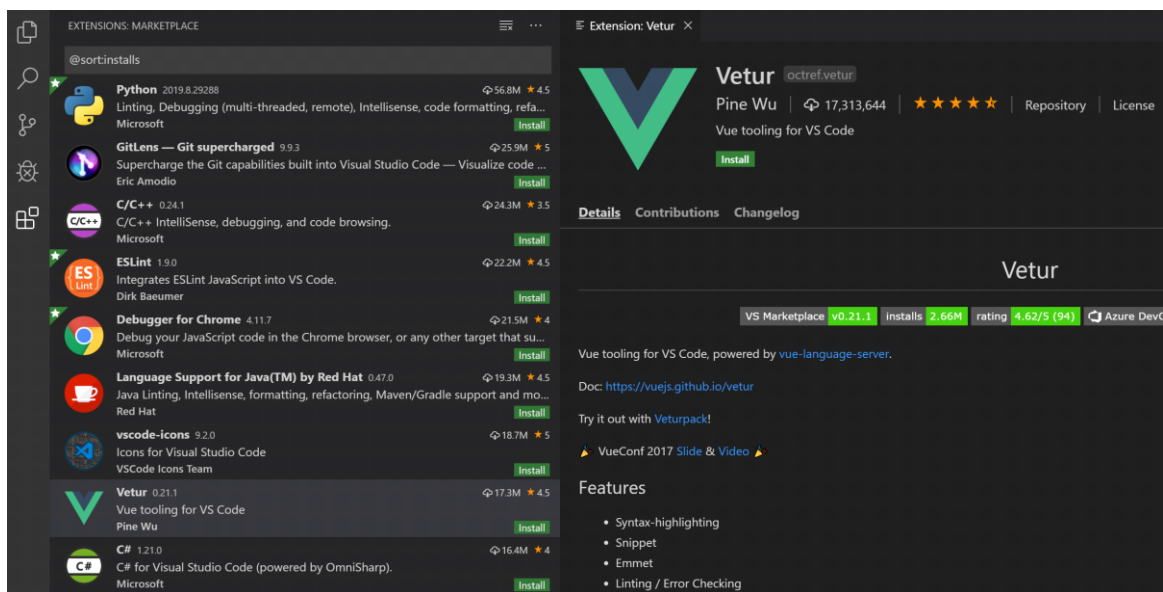


Рис.4.1. Розширення VS code

Кожне розширення у списку включає короткий опис, видавця, кількість завантажень та п'ять зірок. Ви можете вибрати елемент розширення, щоб відобразити сторінку деталей розширення, де ви можете дізнатись більше.

Visual Studio Code - це легкий, але потужний редактор вихідного коду, який працює на вашому робочому столі та доступний для Windows, macOS та Linux. Він постачається із вбудованою підтримкою JavaScript, TypeScript та Node.js і має багату екосистему розширень для інших мов (таких як C++, C#, Java, Python, PHP, Go) та середовищ виконання (таких як .NET та Unity).

Використовуйте топові розширення, відкрийте додаткові мови, теми, налагоджувач, команди тощо. Зростаюче співтовариство VS Code ділиться своїм секретами, щоб покращити ваш робочий процес.

Перші кроки

Щоб отримати максимальну віддачу від Visual Studio Code, спершу перегляньте кілька вступних тем:

Налаштування - Встановіть код VS для вашої платформи та налаштуйте набір інструментів відповідно до ваших потреб у розробці.

Інтерфейс користувача - Ознайомлення з основним інтерфейсом користувача, командами та функціями редактора коду VS.

Налаштування - Налаштуйте код VS для того, як вам подобається працювати.

Мови - Дізнайтеся про підтримку коду VS для ваших улюблених мов програмування.

Node.js - Цей посібник дозволяє швидко запустити та налагодити веб-програму Node.js.

Azure - код VS чудово підходить для розгортання веб-програм у хмарі.

API розширення - Дізнайтеся, як написати розширення VS Code.

Мовна підтримка

Visual Studio Code включає базову підтримку для найбільш поширених мов програмування. Ця основна підтримка включає виділення синтаксису, відповідність дужок, складання коду та конфігуровані фрагменти. Visual Studio Code також постачається з IntelliSense для JavaScript, TypeScript, JSON, CSS та HTML, а також підтримкою налагодження для Node.js. Підтримку додаткових мов можуть надати вільно доступні розширення на VS Code Marketplace.

Гарячі клавіші Підвищте свою продуктивність за допомогою комбінацій клавіш VS Code.

Збір даних

Visual Studio Code збирає дані про використання та надсилає їх корпорації Майкрософт, хоча це можна вимкнути. Крім того, через відкритий код програми,

телеметричний код доступний для громадськості, яка може бачити, що саме зібрано. За даними корпорації Майкрософт, дані передаються підконтрольним Майкрософт філіям та дочірнім компаніям, хоча правоохоронні органи можуть вимагати їх як частину юридичного процесу.

Контроль версій

Це вбудована функція Visual Studio Code. Він має спеціальну вкладку в рядку меню, де можливо отримати доступ до налаштувань контролю версій та переглянути зміни, внесені до поточного проекту. Щоб використовувати цю функцію, ви повинні зв'язати Visual Studio Code з будь-якою підтримуваною системою контролю версій (Git, Apache Subversion, Perforce тощо). Це дозволяє створювати сховища, а також робити запити push і pull безпосередньо з програми Visual Studio Code.

В опитуванні переповнення стеків 2016 року розробники Visual Studio Code посіли 13 місце серед найкращих популярних інструментів розробки, використовуючи лише 7,2% з 46 613 респондентів. Однак через два роки Visual Studio Code зайняв перше місце, використовуючи 34,9% із 75 398 респондентів. В останньому опитуванні, опитуванні розробників 2019 року, Visual Studio Code все ще посідає перше місце, і його використовують 50,7% з 87 317 респондентів.

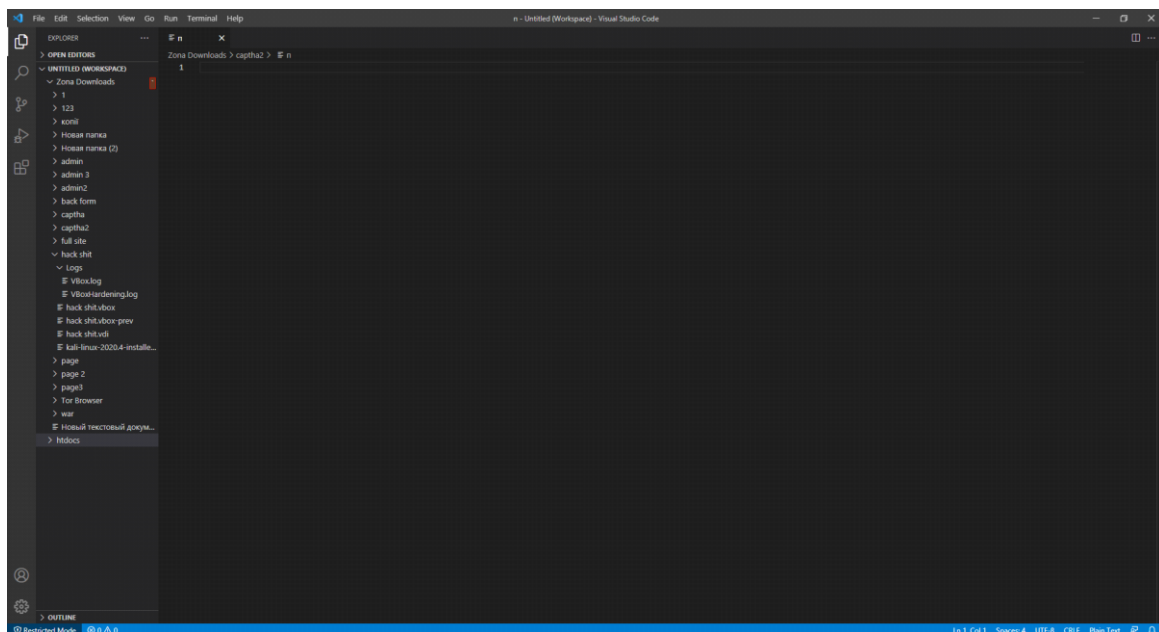


Рис.4.2. Порожній файл робочого місця Microsoft Visual Studio Code

4.2. Програма MAMP

MAMP, рис.3.3 встановлює середовище локального сервера на вашому комп'ютері з ОС Windows або macOS. MAMP поставляється безкоштовно та легко встановлюється. MAMP не пошкодить будь-яку інсталяцію Apache, яка вже запущена у вашій системі. Ви можете встановити Apache, Nginx, PHP та MySQL, не запускаючи скрипт і не змінюючи файли конфігурації!

У режимі CGI можливо запускати кожного з віртуальних хостів із власною версією PHP. Це дозволяє протестувати свої проекти з різними версіями PHP - без необхідності перезапускати сервери. Можете вибрати з різних версій PHP від PHP 5.3.x до PHP 7.3.x.

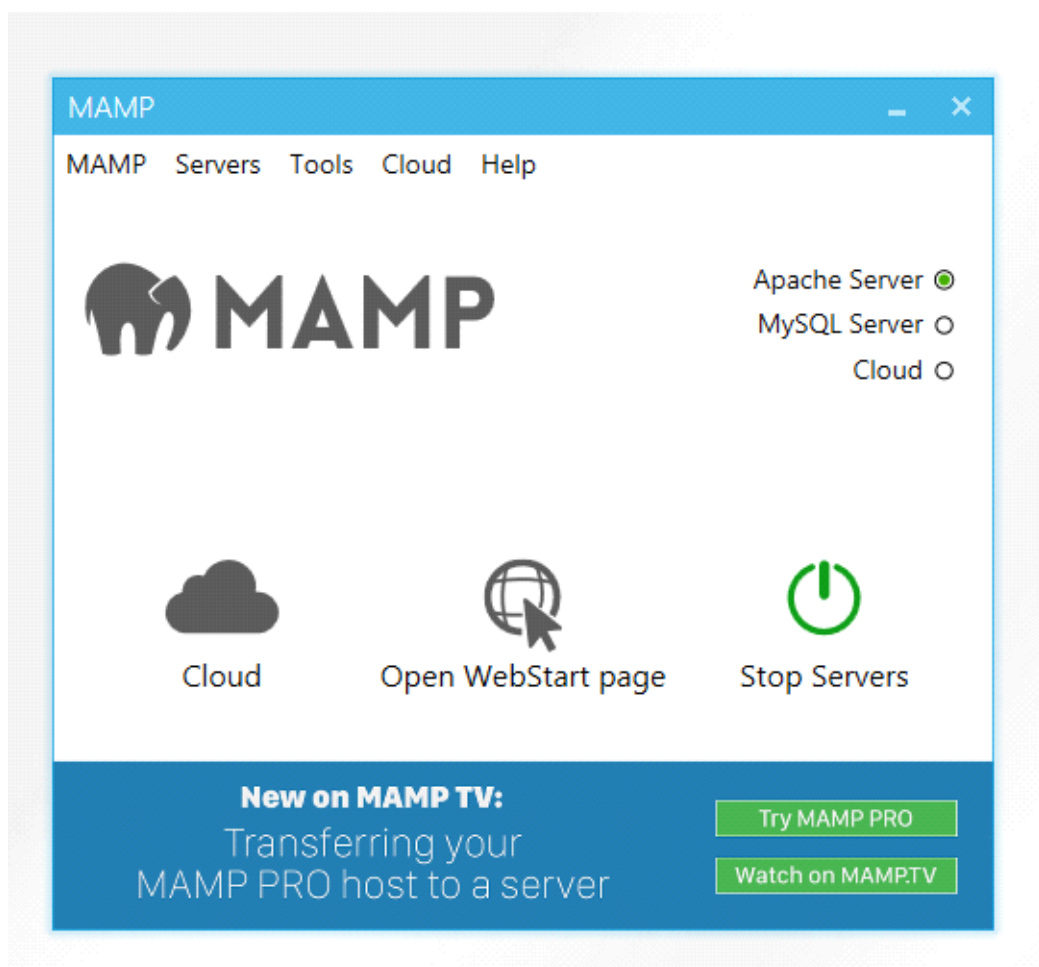


Рис. 4.3. Програма MAMP

У модульному режимі можливо вибрати версію PHP, яка автоматично запускатиметься на всіх ваших хостах. Це корисно, наприклад, якщо ви хочете перевірити, чи працюють ваші проекти під певною версією PHP.

За допомогою функції Cloud можна скопіювати всі або окремі хости у свій буфер. Резервне копіювання кореневого каталогу хосту та його баз даних. Тепер можливо без проблем працювати над своїми проектами на різних комп'ютерах. Також існує можливість використовувати хмарну функцію для резервного копіювання окремих або всіх хостів. На відміну від безкоштовного MAMP, хмарна функція вже включена в MAMP PRO. Тестування своїх проектів на мобільних пристроях у локальній мережі можливе - завдяки програмі MAMP Viewer для iOS та Android. Він доступний безкоштовно в App Store та Google Play Store. Незалежно від того, чи хочете зробити просту зміну одного зі своїх проектів, або запустити абсолютно новий: за допомогою текстового редактора MAMP PRO можете зробити це, локально або віддалено на сервері. Редактор має всі сучасні функції, такі як автозаповнення, підсвічування синтаксису тощо.

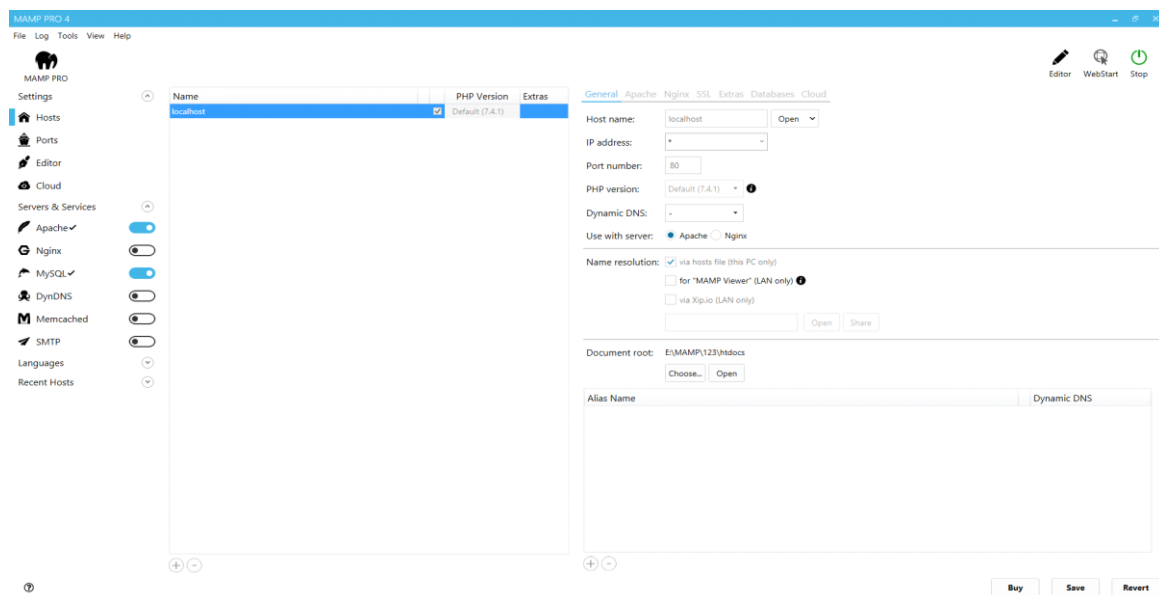


Рис. 4.4. Основна сторінка програми MAMP PRO

Додатковою є пакет веб-програми або бібліотеки, які попередньо протестували та налаштували в програмі MAMP PRO. Коли створюєте

віртуальний хост, все, що потрібно зробити, це вибрати потрібний додатковий файл, і все буде встановлено автоматично. Додаткові функції також можна встановити на існуючі хости.

За допомогою MAMP PRO можливо легко підключити локальний сервер до існуючого динамічного постачальника DNS, наприклад `no-ip.com`. Це дозволяє клієнтам та зацікавленим сторонам дуже швидко отримати доступ до результатів вашої роботи через Інтернет. Ще раз, можливість налаштування будь-якої кількості віртуальних хостів є переконливою перевагою даного програмного забезпечення. Зараз в програмі є величезний вибір постачальників послуг Дуп-DNS, багато з яких безкоштовні.

Назва MAMP - це скорочення, що походить від назв компонентів системи: macOS (операційна система); Apache (веб-сервер); MySQL або MariaDB (система управління базами даних); і PHP, Perl або Python (мови програмування, що використовуються для веб-розробки). Назва походить від LAMP, подібного стеку всього програмного забезпечення з відкритим кодом, що широко використовується для веб-сайтів, але замінюючи власний macOS на ОС Linux з відкритим кодом. (Подібні стеки "AMP" існують і для інших операційних систем.) Однак MAMP не обмежується цими виборами компонентів; Наприклад, замість Apache можна використовувати Nginx, те саме стосується заміни MariaDB на MySQL.

Деякі програмні пакети, що містять MAMP (зокрема Apache та PHP), попередньо встановлені з macOS; сумісні версії решти легко доступні для встановлення та використання. MAMP зазвичай використовується з і розробляється для таких популярних програм управління вмістом, як WordPress та Drupal, шляхом налаштування локального середовища розробки на портативних або настільних комп'ютерах [8].

4.3. Послідовний опис розробки веб орієнтованої інформаційної системи

Спершу була створена персоналізована сервісна сторінка підтримки одиниці техніки. На даній сторінці викладач, інженер, студент має можливість залишити відгук про виявлену поломку і відправити його відповідальному за ремонт техніки, рис.4.5.

На рисунку 4.5 видно що прилад є у базі даних. Адже перші 4 стрічки такі як Кафедра, номер аудиторії та інші були взяті з бази

Кафедра: ІТ
Аудиторія: 56
Одиниця: проектор
Інвентарний номер: 23524
Відповідальний: Олексій А.А.
Заповнити

Повідомити про вирішення проблеми?

Рис. 4.5. Унікальна персоналізована сервісна сторінка підтримки одиниці техніки

Кафедра: ІТ
Аудиторія: 44
Одиниця: Системний блок
Інвентарний номер: 42512
Відповідальний: Олександр К.С.
Заповнити

The image shows a web form for reporting a problem. It contains the following elements from top to bottom: a text input field with the placeholder 'Ваше ім'я'; a larger text area with the placeholder 'описіть проблему'; a checked checkbox with the label 'Повідомити про вирішення проблеми?'; a text input field with the placeholder 'Email-адрес'; another text input field with the placeholder 'Ваш номер телефону'; a grey button with the number '904892' inside; and finally, a light grey button with the text 'відправити'.

Рис.4.6 Унікальна персоналізована сервісна сторінка з функцією hidden

На рисунку 4.6 можна помітити функцію hidden яка приховує поля вводу емейл адреси користувача та поле для вводу номера телефону до поки користувач не натисне на галочку про прохання повідомити коли проблема буде вирішена. Після вводу емейл адреси та номера телефону відповідальному інженеру на поштову скриньку прийде лист з ім'ям користувача описом проблеми та контактними даними для зворотнього зв'язку.

А також був введений комп'ютерний тест у вигляді зображення з рандомно генеруючими цифрами та буквами. CAPTCHA тест типу виклик-відповідь, який використовується для того, щоб визначити, хто використовує систему – людина чи комп'ютер. Для запобігання бот-атак. Щоби пройти його

потрібно ввести букви та цифри з зображення в поле для капчі в іншому випадку система не дасть виконати відправку повідомлення.

```

htdocs > image.php
1  <?php
2
3  //image.php
4
5  session_start();
6
7  $random_alpha = md5(rand());
8
9  $captcha_code = substr($random_alpha, 0, 6);
10
11  $_SESSION['captcha_code'] = $captcha_code;
12
13  header('Content-Type: image/png');
14
15  $image = imagecreatetruecolor(200, 38);
16
17  $background_color = imagecolorallocate($image, 192, 192, 192);
18
19  $text_color = imagecolorallocate($image, 255, 255, 255);
20
21  imagefilledrectangle($image, 0, 0, 200, 38, $background_color);
22
23  $font = dirname(__FILE__) . '/Arial.ttf';
24
25  imagettftext($image, 20, 0, 60, 28, $text_color, $font, $captcha_code);
26
27  imagepng($image);
28
29  imagedestroy($image);
30
31  >

```

Рис.4.7. Створення та налаштування зображення капчі

Завдяки властивості `$font` було завантажено шрифт Arial для відображення тексту на зображенні.

`$font`-Універсальна властивість, яка дозволяє задати кілька характеристик шрифту і тексту.

Також на рис. 4.7 можна побачити налаштування кольору тексту та фону за допомогою властивостей `background-color` і `text-color`.

```

<script>
$(document).ready(function(){

$('#captch_form').on('submit', function(event){
event.preventDefault();
if($('#captcha_code').val() == '')
{
alert('Enter Captcha Code');
$('#register').attr('disabled', 'disabled');
return false;
}
else
{
alert('Form has been validate with Captcha Code');
$('#captch_form')[0].reset();
$('#captcha_image').attr('src', 'image.php');
}
});

$('#captcha_code').on('blur', function(){
var code = $('#captcha_code').val();

if(code == '')
{
alert('Enter Captcha Code');
$('#register').attr('disabled', 'disabled');
}
else
{
$.ajax({
url:"check_code.php",
method:"POST",
data:{code:code},
success:function(data)
{
if(data == 'success')
{
$('#register').attr('disabled', false);
}
else
{
$('#register').attr('disabled', 'disabled');
alert('Invalid Code');
}
}
});
}
});
});

```

Рис.4.8. Скрипт який виконує перевірку правильності капчі

```

1  <?php
2
3  //check_code.php
4
5  session_start();
6
7  $code = $_POST['code'];
8
9  if($code == $_SESSION['captcha_code'])
10 {
11     echo 'success';
12 }
13
14 >
```

Рис.4.9. Перевірка капчі на коректність

Після введення даних користувачем та проведену перевірку людина/робот на поштову скриньку відповідальної за ремонт особи приходить лист з інформацією яку написав користувач.

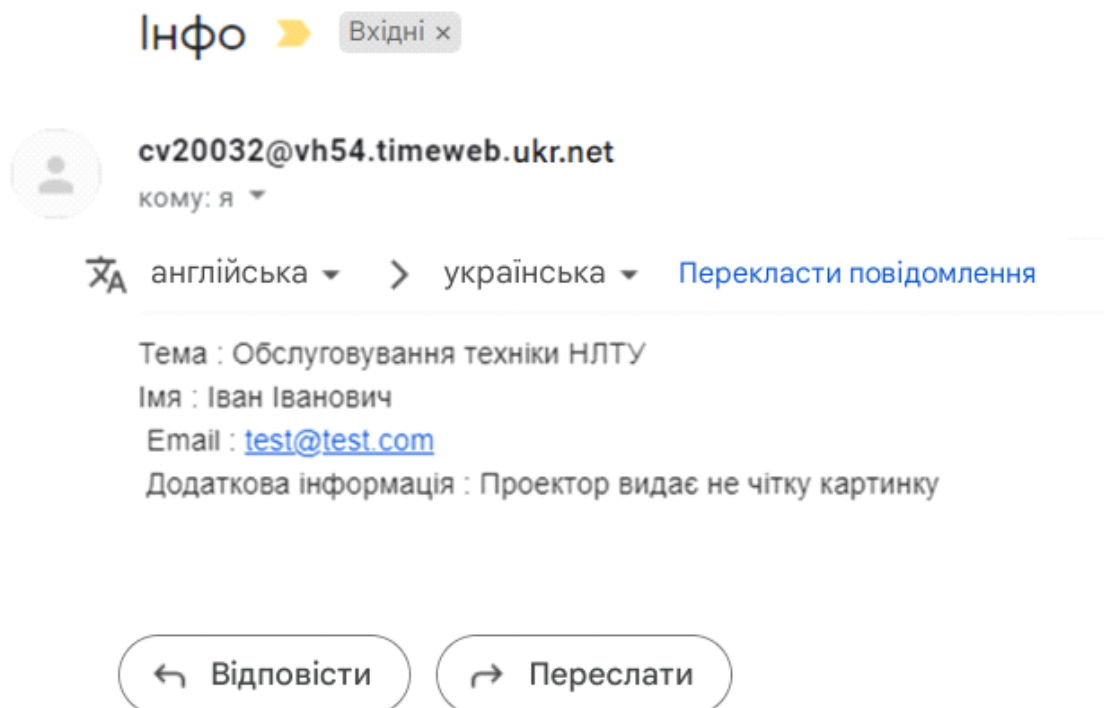


Рис.4.10. Приклад листа надісланого користувачем до адміністратора

```

if (isset($name1) ) {
$name1=stripslashes($name1);
$name1=htmlspecialchars($name1);
}
if (isset($email1) ) {
$email1=stripslashes($email1);
$email1=htmlspecialchars($email1);
}
if (isset($text) ) {
$text=stripslashes($text);
$text=htmlspecialchars($text);
}
// адреса електронної пошти куди прийде лист
$address="test@gmail.com";
// текст листа
$note_text="Тема : $urok \r\nІмя : $name1 \r\n Email : $email1 \r\n Додаткова інформація : $text";

if (isset($name1) && isset ($sab) ) {
mail($address,$urok,$note_text,"Content-type:text/plain; windows-1251");
}

```

Рис.4.11 Программний код який відповідає за надсилання повідомлення на пошту адміністратора

В функції \$address потрібно вписати поштову скриньку на яку буде приходити дані від користувачів. Далі було реалізовано сервісну сторінку адміністратора де відповідальний за ремонт працівник міг би проводити інвентаризацію. Переглядати та за необхідності змінювати або добавляти чи віднімати одиниці техніки в кожному кабінеті. Для входу на сторінку адміністратора потрібно буде знати пароль та логін.

Логін

Введіть свій логін

Пароль

Введіть свій пароль

Увійти

Рис.4.12 Панель авторизації для адміністратора

Після авторизації адміністратор попаде на головну сторінку сайту.

Сервісне обслуговування техніки

Виберіть кафедру ▲
ІТ
Бібліотека
Маркетинг ▼

Добавити

Виберіть аудиторію ▲
56
44
42 ▼

Добавити

Відповідальний за ремонт

Олександр К.С. ▼ Поштова скринька (Oleksandr.K@ua.ua)

Рис.4.13 Сторінка адміністратора

На цій сторінці можна буде добавляти техніку в будь який кабінет натиснувши кнопку добавити дані в таблицю.

Рис.4.14 Спливаюче вікно для заповнення даних після натискання кнопки
добавити дані в таблицю

Дата	Година	Опис проблеми	Інвентарний номер	Відповідальний	Відмітка про виконання
21.12.22	10:20	Проектор видає нечітке зображення	4570	Олексій А.А.	В роботі
20.12.22	14:10	Системний блок видає дивні звуки	2790	Олександр К.С.	Виконано

Рис.4.15 Таблиця виконання завдань

На рисунку 4.15 зображена таблиця із завданнями які надходили від користувачів на протязі останнього часу. На цій таблиці ми можемо бачити дату та годину коли було отримане завдання опис проблеми з електронного листа

інвентарний номер одиниці техніки відповідального інженера аудиторії в якій знаходиться проблемний прилад та хід виконання завдання.

Всі дані будуть зберігатися в базі даних кафедри ІТ. Для демонстрації була використана локальна база даних програми МАРМ .

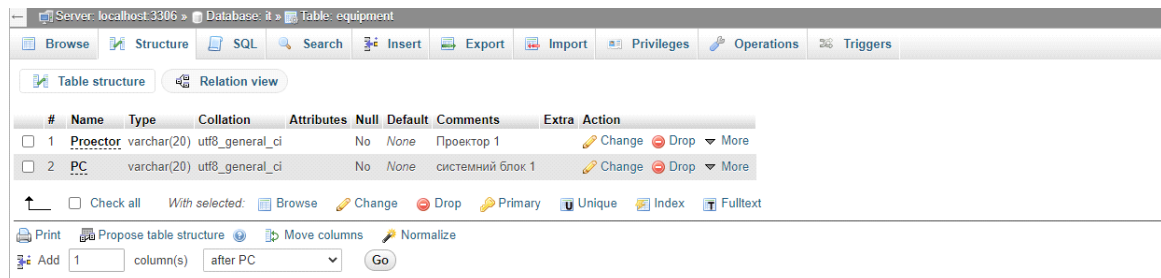


Рис.4.16 Панель управління базами даних phpmyadmin

Дані можна буде добавляти до бази даних вручну через панель phpAdmin рис.4.16 або ж тиснути на кнопку “добавити дані в таблицю” на сторінці адміністратора рис.4.13. Дані автоматично будуть додаватися до бази даних.

Для генерування QR-code було використано фреймворк Angular та мову програмування JavaScript для створення унікального qr-code для кожної одиниці техніки. В середині буде зашифровано посилання на форму для користувача рис.4.6.



Рис.4.17 Сторінка генерації qr-code

```

<!DOCTYPE html>
<html>
<head>
<title>QRCode test</title>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=no" />
<script type="text/javascript" src="jquery.min.js"></script>
<script type="text/javascript" src="qrcode.js"></script>
</head>
<body>
<input id="text" type="text" value=" Введіть посилання" style="width:80%" /><br />
<div id="qrcode" style="width:200px; height:200px; margin-top:15px;"></div>

<script type="text/javascript">
var qrcode = new QRCode(document.getElementById("qrcode"), {
    width : 200,
    height : 200
});

function makeCode () {
    var elText = document.getElementById("text");

    if (!elText.value) {
        alert("Input a text");
        elText.focus();
        return;
    }

    qrcode.makeCode(elText.value);
}

makeCode();

$("#text").
    on("blur", function () {
        makeCode();
    }).
    on("keydown", function (e) {
        if (e.keyCode == 13) {
            makeCode();
        }
    });
</script>
</body>

```

Рис.4.18 Реалізація html коду

```

> 1 322
> 123 323
> копії 324
> Новая папка 325
▼ Новая папка (... 326
  > e2e 327
  > node_modules 329
  ▼ src 330
    ▼ app 331
      TS app-routing.mo... 332
      # app.component.... 333
      <> app.comp... M 334
      TS app.component.... 335
      TS app.component.ts 336
      TS app.module.ts 337
      TS app.module.ts 338
      JS jquery.min.js U 339
      JS qrcode.js U 340
      > assets 342
      > environments 343
      ★ favicon.ico 344
      <> index.html 345
      TS main.ts 346
      TS polyfills.ts 347
      # styles.css 348
      TS test.ts 349
      .editorconfig 350
      .gitignore 351
      {} angular.json M 353
      ≡ browserslist 354
      📄 karma.conf.js 355
      {} package-lock.json 356
      {} package.json 357
      ⓘ README.md 358
      {} tsconfig.app.json 360
      📄 tsconfig.json 361
      {} tsconfig.spec.json 362
      {} tslint.json 363
      OUTLINE 365
      TIMELINE 366
      NPM SCRIPTS 367
  
```

```

* малюнок QR-код використовуючи canvas
*
* @constructor
* @param {HTMLElement} el
* @param {Object} htOption QRCode Options
*/
var Drawing = function (el, htOption) {
  this._bIsPainted = false;
  this._android = _getAndroid();

  this._htOption = htOption;
  this._elCanvas = document.createElement("canvas");
  this._elCanvas.width = htOption.width;
  this._elCanvas.height = htOption.height;
  el.appendChild(this._elCanvas);
  this._el = el;
  this._oContext = this._elCanvas.getContext("2d");
  this._bIsPainted = false;
  this._elImage = document.createElement("img");
  this._elImage.alt = "Scan me!";
  this._elImage.style.display = "none";
  this._el.appendChild(this._elImage);
  this._bSupportDataURI = null;
};

/**
 * малюнок QR-код
 *
 * @param {QRCode} oQRCode
 */
Drawing.prototype.draw = function (oQRCode) {
  var _elImage = this._elImage;
  var _oContext = this._oContext;
  var _htOption = this._htOption;

  var nCount = oQRCode.getModuleCount();
  var nWidth = _htOption.width / nCount;
  var nHeight = _htOption.height / nCount;
  var nRoundedWidth = Math.round(nWidth);
  var nRoundedHeight = Math.round(nHeight);

  _elImage.style.display = "none";
  this.clear();

  for (var row = 0; row < nCount; row++) {
  
```

Рис.4.19 Використання елементу canvas

Елемент canvas надає можливість використовувати скрипти для малювання графіки.

```

350  * @param {QRCode} oQRCode
351  */
352  Drawing.prototype.draw = function (oQRCode) {
353      var _elImage = this._elImage;
354      var _oContext = this._oContext;
355      var _htOption = this._htOption;
356
357      var nCount = oQRCode.getModuleCount();
358      var nWidth = _htOption.width / nCount;
359      var nHeight = _htOption.height / nCount;
360      var nRoundedWidth = Math.round(nWidth);
361      var nRoundedHeight = Math.round(nHeight);
362
363      _elImage.style.display = "none";
364      this.clear();
365
366      for (var row = 0; row < nCount; row++) {
367          for (var col = 0; col < nCount; col++) {
368              var bIsDark = oQRCode.isDark(row, col);
369              var nLeft = col * nWidth;
370              var nTop = row * nHeight;
371              _oContext.strokeStyle = bIsDark ? _htOption.colorDark : _htOption.colorLight;
372              _oContext.lineWidth = 1;
373              _oContext.fillStyle = bIsDark ? _htOption.colorDark : _htOption.colorLight;
374              _oContext.fillRect(nLeft, nTop, nWidth, nHeight);
375
376              _oContext.strokeRect(
377                  Math.floor(nLeft) + 0.5,
378                  Math.floor(nTop) + 0.5,
379                  nRoundedWidth,
380                  nRoundedHeight
381              );
382
383              _oContext.strokeRect(
384                  Math.ceil(nLeft) - 0.5,
385                  Math.ceil(nTop) - 0.5,
386                  nRoundedWidth,
387                  nRoundedHeight
388              );
389          }
390      }
391
392      this._bIsPainted = true;
393  };
394
395  Drawing.prototype.makeImage = function () {
396      if (this._bIsPainted) {

```

Рис.4.20 Частина скрипта який створює зображення Qr-code

Висновки до розділу

Результатом виконання даного розділу є спроектована архітектура системи. Після аналізу предметної області майбутньої системи було вирішено використовувати клієнт-серверну архітектуру. Важливим етапом в проектуванні програмного продукту було створення генерацію к'ю ар коду з посиланням на форму для користувача. Також розроблено простий та зручний інтерфейс користувача, який дозволяє швидко і просто повідомити відповідальну людину про проблему . З допомогою цього прототипу взаємодія між користувачем та адміністратором відбувається значно швидше.

Цей розділ є надзвичайно важливим, оскільки допущені помилки на даному етапі призведуть, до суттєвих змін в подальшому у розробленій системі. Після всіх етапів проектування було реалізовано систему на основі вищезгаданих архітектур, бази даних та прототипу. Також в цьому розділі представлена робота інформаційної веб-системи з відображенням всього функціоналу.

РОЗДІЛ 5. РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ

5.1. Опис ідеї проекту

Даний проект створений для того, щоб пришвидшити та полегшити передачу інформації щодо певних проблем з технічним обладнанням навчального закладу шляхом автоматизації передачі цієї інформації. Завдяки даній системі користувачі зможуть повідомляти про несправності просто відсканувавши QR-код. Окрім цього представлений функціонал для робітників навчального закладу, які зможуть генерувати QR-коди для кожної одиниці обладнання, оперативно отримувати email-повідомлення, в якому буде описана проблема та призначати відповідного за вирішення несправностей.

Розробка призначена для того, щоб користувачі могли оперативно повідомляти про проблеми, що виникли з технічним обладнанням, а робітники отримувати точну інформацію про те, яка саме одиниця техніки потребує ремонту. Дана система дозволить зекономити час як користувачам, так і робітникам навчального закладу.

Створений продукт допоможе працівникам навчального закладу покращити якість надання своїх послуг, адже вони зможуть швидко отримувати повідомлення про проблему та реагувати на неї. Окрім цього користувачі за бажання можуть отримати сповіщення про завершення ремонту, що значно підвищить рівень зацікавленості та задоволення користувача.

Значною перевагою даної розробки є спосіб, за допомогою якого користувач може повідомити про проблему. Для цього достатньо лише відсканувати QR-код, розміщений на приладі, та ввести текстове повідомлення, у якому розповісти про несправності. Для зручності користувача, у робочому вікні він буде бачити назву одиниці техніки, кабінет, у якому розміщена ця техніка, що дозволить споживачеві зорієнтуватись у тому, чи правильну

інформацію він надає та зекономити час, адже не виникне потреби вносити дану інформацію самостійно.

При розробці даної системи було враховано можливість спаму інтернет-ботів, тож для того, щоб полегшити роботу працівникам навчального закладу, було використано систему захисту типу CAPTCHA. Перед тим як користувач надішле повідомлення про несправності в роботі технічного засобу, йому потрібно буде пройти цю перевірку.

Під час аналізу подібних програм на ринку, не було виявлено аналогів, що надає даній розробці значної переваги. Інновація полягає саме у QR-коді, за допомогою якого користувач відразу відкриває вікно, де можна повідомити про проблему.

Отже, дана розробка буде дуже корисною для роботи навчального закладу (або ж будь-якої компанії), адже вона дозволяє зекономити час працівнику та користувачеві завдяки автоматизації способу повідомлення про проблему, що виникла з технічним обслуговуванням.

Користувач зможе швидко надавати інформацію про проблему, адже завдяки унікальним QR-кодам, йому не потрібно вносити інформацію про одиницю техніки, її розташування і т.д., а лише заповнити поле з описом проблеми. Також користувач за бажання може отримати оповіщення про те, що проблему було вирішено.

Для робітника ж система буде корисною тим, що він отримувати точну інформацію щодо одиниці техніки, завдяки ж таму-таки QR-коду. Крім того, завдяки використанню системи CAPTCHA, робітників захищений від спаму інтернет-ботів.

5.2. Аналіз технологічних можливостей реалізації ідей проекту

Перше завдання реалізації даного етапу стартапу – це провести технологічний аудит, завдяки якому можливо реалізувати усі основні ідеї проекту.

Таблиця 5.1 Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1.	Розроблення <u>веб-орієнтованої інформаційної системи</u> “Сервісне обслуговування техніки кафедри ІТ НЛТУ України” засобами <u>AngularJS</u>	HTML	Є у наявності	Безкоштовна. Відкрита для всіх операційних систем
		CSS	Є у наявності	Безкоштовна. Відкрита для всіх операційних систем
		<u>Java script</u>	Є у наявності	Безкоштовна. Відкрита для всіх операційних систем
		<u>AngularJS</u>	Є у наявності	Безкоштовна. Відкрита для всіх операційних систем
Обрана технологія реалізації проекту: <u>Java</u> + <u>REST</u> + <u>Spring Boot</u> + <u>PostgreSQL</u>				

Отже, як можна побачити з таблиці 5.1, реалізація ідеї продукту має право на успіх, адже технологічна здійсненність має великий відсоток успіху. Технології, які необхідні для проектування та реалізації задуманої системи цілком безкоштовні та доступні кожному розробнику.

5.3. Аналіз ринкових можливостей запуску стартап-проекту

Для того, аби визначити ринкові можливості запуску даного стартапу, проведено аналіз попиту, що включає кількість основних представників на ринку одиниць, загальний обсяг продажів для сектора, динаміку ринку, наявність обмежень для входу на ринок, специфічні вимоги до стандартизації та сертифікації і середню норму рентабельності.

Таблиця 5.2 Аналіз ринкових можливостей запуску стартап-проекту

№ п/п	Показники стану ринку	Характеристика
1.	Кількість основних представників на ринку, одиниць	5-10
2.	Загальний обсяг продажів для сектора	1,5 млрд. доларів
3.	Динаміка ринку за якісною оцінкою	Зростаюча
4.	Наявність обмежень для входу на ринок	Невелика кількість аналогів на ринку та невелика проте значна конкуренція
5.	Специфічні вимоги до стандартизації та сертифікації	Не потребує
6.	Середня норма рентабельності в галузі, %	R = 26%

Аналіз попиту

У таблиці 5.3 визначено та охарактеризовано потенційні групи клієнтів і сформовано перелік вимог до продукту для кожної групи клієнтів.

Як бачимо, цільова аудиторія досить обмежена і в основному система розрахована на користувачів та працівників, що не є позитивним аспектом стартапу.

Після визначення та характеристики потенційних груп клієнтів доцільно провести аналіз ринкового середовища, що в свою чергу передбачає визначення основних факторів, що сприяють ринковому впровадженню проекту (фактори можливостей) та факторів загроз, що перешкоджають реалізації проекту на ринку.

Фактори, що сприяють ринковому впровадженню проекту:

1. Перш за все, система дуже зручна для користувачів: просто відсканувавши GR-код, вони одразу можуть перейти до вікна, де можна лишити повідомлення про несправності та за бажання отримати зворотну реакцію;
2. Розробка створена для того, щоб полегшити роботу модераторів сайту, адже, по-перше, вони захищені від спаму інтернет-ботів завдяки системі CAPTCHA, а по-друге, завдяки унікальному QR-коду, який присвоюється

кожній одиниці техніки, працівник може швидко, легко і точно ідентифікувати одиницю техніки, з якою виникли проблеми;

3. Оформлення інтерфейсу та каталогу здійснено на професійному рівні, що в свою чергу забезпечує зручну роботу з програмою.

Таблиця 5.3 Характеристика потенційних клієнтів стартап-проекту

№ п/п	Цільова аудиторія	Відмінності у поведінці різних потенційних цільових груп клієнтів	Потреба, що формує ринок	Вимоги споживачів до продукту
1.	Звичайні студенти, викладачі які хочуть повідомити про проблему	Різні методи прийняття рішень та різні види несправностей	Облік технічного забезпечення кафедри в одній базі	Можливість оперативно доповісти про проблему відповідальному лицю, та допомогти коментарем з описом проблеми
2.	Системний адміністратор	Спрощення комунікації між користувачем та виконавцем і статистики технічний стан обладнання	Швидкість виявлення проблеми та якнайшвидше її усунення	Реагування на технічні несправності які виникають та облік і аналіз технічного забезпечення
3.	Інші гості університету	Виявити та доповісти про проблему може будь який відвідувач.	Простота використання	Можливість ознайомлення із системою обслуговування техніки кафедри

Таблиця 5.4 Фактори загроз

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1.	Розробка	Постійний контроль зі сторони системного адміністратора	Здійснення пошуку та реалізації різноманітних альтернативних варіантів та компонентів
2.	Ринок	вужько направлена інформаційна система з безліччю аналогів	Адаптація до умов ринку та його розширення (перш за все локальне)
3.	Фінанси	Збільшення витрат на відповідальних інженерів які виконують ремонт	Залучення інвестицій, участь у партнерських програмах
4.	Конкуренція	Середня кількість конкурентів і, відповідно, середня конкуренція на ринку	Постійне підвищення якості продукту і забезпечення якнайкращого співвідношення ціна/якість

Варто також проаналізувати рівні та види конкуренції на ринку (див. табл. 5.5).

Таблиця 5.5 Аналіз конкуренції на ринку

Особливості конкурентного середовища	Прояв характеристики	Вплив на діяльність підприємства та можливі дії компанії, щоб бути конкурентоспроможною
За інтенсивністю: не марочна	Відсутність впливу бренду на впізнаваність продукції та товарів.	Можливість заключення угод для частково співпраці з брендовими компаніями без взяття логотипу чи повний перехід до брендової компанії.
Конкуренція за видами товарів: товарно-видова	Види товарів є схожими за функціоналом.	Врахування усіх недоліків алгоритмів.
Тип конкуренції: чистий	Вільне конкурування на ринку.	Принцип чесною конкуренції.
За характером конкурентних переваг: цінова	Ціна є одним з найбільш важливих факторів при виборі продукту і впливає на купівельне рішення клієнта.	Більша концентрація та зосередження уваги на вартість розробки і впровадження продукту.
За рівнем конкурентної боротьби: міжнародний	На ринку представлені іноземні представництва-конкуренти.	Додавання до функціоналу можливості вибору мови ПЗ, для легшого виходу на міжнародний ринок у майбутній перспективі. Вивчення пропозицій іноземних фірм-конкурентів для вдосконалення і покращення набутих навиків.
За галузевою ознакою: міжгалузева	Використання технології в багатьох галузях.	Підтримка ідей та проектів щодо створення когнітивних карт, а також пошук нових ідей.

У таблиці 5.6 представлено аналіз сильних та слабких сторін проекту, на основі яких надалі можна навести SWOT - аналіз стартап-проекту (див. табл. 5.7).

Таблиця 5.6 Порівняльний аналіз сильних та слабких сторін проекту

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з проектом							
			-3	-2	-1	0	+1	+2	+3	
1.	Досвід користувача	15								+
2.	Інтерфейс	10							+	
3.	Ціна	20				+				
4.	Зручність користування	18							+	

SWOT-аналіз є методом оцінки внутрішніх і зовнішніх факторів, що значно впливають на розвиток проекту чи компанії. Завдяки цьому аналізу можна виявити сильні і слабкі сторони проекту, знайти та охарактеризувати можливості і загрози.

Таблиця 5.7 SWOT - аналіз стартап-проекту

Сильні сторони: Зручність у користування з великої кількості пристроїв, досвід користувача, інтерфейс, інноваційність.	Слабкі сторони: Недосконалий метод вимірювання об'єктів.
Можливості: Вияв високої зацікавленість серед груп клієнтів і споживачів пропонуваним продуктом	Загрози: Невелика цільова аудиторія.

5.4. Розроблення ринкової стратегії проекту

Етап розроблення ринкової стратегії проекту передбачає визначення стратегії охоплення ринку, а саме: опис цільових груп потенційних споживачів продукту (див. табл. 5.8), визначення базових стратегій розвитку проекту (див. табл. 5.9) та визначення базової стратегії конкурентної поведінки (див. табл. 5.10)

Таблиця 5.8 Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи	Готовність споживачів сприйняти продукт	Орієнтований попит в межах сегменту	Інтенсивність конкуренції в сегменті	Простота входу в сегмент
1.	Користувачі та клієнти додатку	Висока готовність	Високий	Середня	Висока
2.	Меблеві магазини	Висока готовність	Високий	Середня	Середня
3.	Інші зацікавлені користувачі	Середня готовність	Високий	Середня	Доступно

Таблиця 5.9 Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1.	Розроблення каталогу меблів із елементами доповненої реальності	Стратегія концентрованого маркетингу	Підвищення якості продукту та функціональних параметрів	Стратегія широкої диференціації

Таблиця 5.10 Визначення базової стратегії конкурентної поведінки

Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
Представлений проект є першопрохідцем на ринку послуг	Незважаючи на доволі велику кількість конкурентів, проект буде мати активні конкурентні позиції	Створення та реалізація проекту передбачає використання власних ідей та рішень, а також використання та звертання до вже існуючих характеристик	Стратегія наслідування лідера (дозволяє уникнути боротьби з лідерами ринку)

5.5. Розроблення маркетингової програми стартап-проекту

Щоб розробити макет маркетингової програми стартап-проекту потрібно визначити основні переваги концепцій потенційного продукту, які отримає споживач та здійснити підсумок результатів попередньо проведеного аналізу конкурентоспроможності продукту.

Таблиця 5.11 Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами
1.	Зворотній зв'язок між користувачем та адміністратором	Можливість зручно та швидко комунікувати між користувачем та відповідальним за ремонт персоналом	Використання тие матричного штрих-коду в якості посилання

продовження таблиця 5.11

2.	Підтримка функціонування постійне підвищення його якості.	Легкість у користуванні, зручний інтерфейс та функціонал для здійснення запиту на огляд та ремонт техніки	Нижча вартість
----	---	---	----------------

Таблиця 5.12 Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові
I. Товар за задумом	Розроблення веб орієнтованої системи для спрощення комунікації між користувачем та виконавцем завдань
II. Товар у реальному виконанні	Наділений властивостями: зручний та простий у навігації інтерфейс та функціонал для здійснення своєчасного виявлення поломок та пришвидшить ремонт техніки
III. Товар із підкріпленням	До продажу: повна версія
	Після продажу: постійне підвищення якості товару та його модернізація
Так, як основне написання програмного коду та створення додатка здійснюються на віддаленому сервері та платформі, товар захищено від копіювання. Окрім того, при написанні програмного коду використано систему безпеки CAPTCHA для захисту від інтернет ботів.	

Таблиця 5.13 Визначення меж встановлення ціни

Рівень цін на товари-замінники	Рівень цін на товари - аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
410-910\$	650-1150\$	10000\$	275-650\$

Після цього потрібно сформувати систему збуту, включає ряд таких критеріїв як Специфіка закупівельної поведінки цільових клієнтів, функції збуту, глибина каналу збуту, оптимальна система збуту.

Таблиця 5.14 Формування системи збуту

Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
Клієнти та споживачі мають на меті отримувати товар за нижчою ціною та за рахунок провадження конкуренції	Продаж товару потенційним клієнтам (надання доступу, можливість користування)	Канали збуту нульового рівня та однорівневий	Прямі канали розподілу (власноруч) та за допомогою посередників

Останнім етапом є концепція маркетингових комунікацій, що опирається на попередньо обрану платформу (основу) для впорядкування та визначену своєрідність (специфіку) поведінки споживачів і клієнтів (див. табл. 5.15).

Таблиця 5.15 Концепція маркетингових комунікацій

Специфіка поведінки цільових клієнтів	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Канали комунікацій	Концепція рекламного звернення
Пошук пропонувань товарів через Інтернет мережу та рекламу	Забезпечення конкурентоспроможного становища продукту на ринку, моніторинг і прогнозування, підвищення якості	Представлення переваг продукту та ефективність його роботи з чітким акцентом на цінове та якісне співвідношення	Телефонний зв'язок, Інтернет-мережа, комунікаційні системи	Зручність та якість за доступною ціною

Висновки до розділу

Завдяки своїй продуманості та орієнтації і на користувача, і на працівника, програму можна вважати успішною. Було проведено аналіз інформаційної системи та можливостей її впровадження та функціонування у вигляді стартап-проекту. Спочатку був зроблений аналіз попиту та потреби потенційних споживачів. Здійснено SWOT-аналіз, після якого можна ознайомитись з сильними та слабкими сторонами проекту, а також виявити можливості та загрози для його існування. Найбільшою загрозою для проекту є обмежене коло користувачів, тому постійно потрібно думати над покращенням функціоналу, щоб залучити все більше зацікавлених користувачів для користування системою.

Підсумовуючи все вище викладене, можна сказати, що програма з легкістю може завоювати місце на ринку, хоч і з обмеженою цільовою аудиторією. Завдяки своїй унікальності та інноваційності система може легко стати однією з провідних та затребуваних у своїй сфері.

ВИСНОВКИ

Розроблено інформаційну систему для сервісного обслуговування техніки кафедри ІТ НЛТУ України. Реалізоване динамічне створення QR-коду із закодованим URL-посиланням на сервісну сторінку підтримки одиниці техніки, на якій викладач, інженер, студент має можливість залишити відгук про виявлену поломку і відправити його відповідальному за ремонт. Для даної сторінки реалізовано тест CAPTCHA, щоб позбавитись спаму від інтернет-ботів. Адміністратор інформаційної системи бачить усі запити щодо виявлених несправностей та відповідальних за їх усунення.

Організовано швидку взаємодію з базою даних, взаємодію з бекенд частиною застосунку та обробку запитів без перезавантаження сторінки за допомогою AJAX та jQuery. Розроблено україномовний, адаптований під більшість розширень екранів, користувацький інтерфейс засобами HTML, PHP та JavaScript.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. White M. Mastering JavaScript: A Complete Programming Guide Including jQuery, AJAX, Web Design, Scripting and Mobile Application Development / Michael B White. – Kindle Edition, 2019
2. John D. HTML & CSS: Design and Build Web Sites / D John. – John Wiley & Sons, Inc., 2018. – 514с
3. Набір інструментів з відкритим кодом, призначений для створення веб-сайтів та веб-додатків [Електронний ресурс] – Режим доступу до ресурсу: getbootstrap.com. – доступний станом на 12.12.2022
4. JavaScript-бібліотека з відкритим кодом [Електронний ресурс] – Режим доступу до ресурсу: api.jquery.com. – доступний станом на 12.12.2022
5. Josh L. Modern PHP: New Features and Good Practices / L Josh. – Published March 1st 2019 by O'Reilly Media
6. Krishna R. Learn Angularjs in 1 Day: Complete Angular Js Guide with Examples / R Krishna, 2018
7. Офіційна документація по Mysql [Електронний ресурс] - Режим доступу до ресурсу: <https://dev.mysql.com/doc/refman/8.0/en>. – доступний станом на 12.12.2022
8. MAMP [Електронний ресурс] – Режим доступу до ресурсу: <https://www.mamp.info/en/mamp-pro/windows>. – доступний станом на 12.12.2022.
9. QR Code Demystified [Електронний ресурс] – Режим доступу до ресурсу: <https://matchadesign.com/blog/qr-code-demystified-part-1/> – доступний станом на 12.12.2022.

ДОДАТКИ

HTML код сторінки користувача

```
<body>
```

```
  <form class="form" method="post"  
action="https://nltudiplom.000webhostapp.com/">
```

```
  <div class="form-row">
```

```
    <div class="form-group col-6">
```

```
      <label for="name" style="font-size:2vw;"><strong>Кафедра: IT</strong>  
</label>
```

```
    </div>
```

```
    <div class="form-group col-6">
```

```
      <label for="mail" style="font-size:2vw;"><strong>Аудиторія: 44</strong>  
</label>
```

```
    </div>
```

```
  </div>
```

```
  <div class="form-group">
```

```
    <label for="phone" style="font-size:2vw;"><strong>Одиниця: Системний  
блок</strong></label>
```

```
  </div>
```

```
  <div class="form-group">
```

```
    <label for="company" style="font-size:2vw;"><strong>Інвентарний номер:  
42512</strong></label>
```

```
  <div class="form-group">
```

```
    <label for="company" style="font-size:2vw;"><strong>Відповідальний:  
Олександр К.С.</strong></label>
```

```

</div></div>
<div class="form-group">
  <label for="website" style="font-size:2vw;"><strong>Заповнити </strong>
</label>
</div>
<div>

  <input type="text" id="name" placeholder="Ваше ім'я">

</div>

  <textarea cols="32" name="message" rows="2" placeholder="22222">
опишіть проблему
</textarea>

  <div class="form-group">
    <input type="checkbox" name="agree" class="custom-control-input"
id="customCheck"> Повідомити про вирішення проблеми?

  <div class="form-group">
    <label for="email" class="control-label"></label>

    <input id="email" type="email" name="email" required="required"
class="form-control" value="" placeholder="Email-адрес">

    <div class="invalid-feedback"></div> <input type="text" id="name"
placeholder="Ваш номер телефону">

  </div><br><input type="submit" value="Відправити">

</div></form>

```

```
</body>
```

CSS код сторінки користувача

```
form{
```

```
    height: 100em;
```

```
    position:relative;
```

```
    left:610px;
```

```
    top:2px
```

```
}
```

```
textarea{
```

```
    display: inline-block;
```

```
    margin: 10px auto;
```

```
    border: 2px solid #eee;
```

```
    padding: 10px 20px;
```

```
    font-family: Verdana, Arial, sans-serif;
```

```
    font-size: 16px;
```

```
}
```

```
input{
```

```
    margin: 10px auto;
```

```
    border: 2px solid #eee;
```

```
    padding: 10px 20px;
```

```
    font-family: Verdana, Arial, sans-serif;
```

```
    font-size: 16px
```

```
}
```

HTML код сторінки адміністратора

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <link rel="stylesheet" href="admins.css">
```

```
  <title>Document</title>
```

```
</head>
```

```
<body>
```

```
  <div>
```

```
    <h3>Сервісне обслуговування техніки</h3>
```

```
    <select multiple="multiple" name="type">
```

```
      <option disabled="disabled" selected="selected">Виберіть кафедру</option>
```

```
      <option value = "Програміст">ІТ</option>
```

```
      <option value = "Системний адміністратор">Бібліотека</option>
```

```
      <option value = "Маркетолог">Маркетинг</option>
```

```
      <br><input type="submit" value="Добавити">
```

```
    </select>
```

```
</div>
```

```
<br> <select multiple="multiple" name="type">
```

```
  <option disabled="disabled" selected="selected">Виберіть аудиторію</option>
```

```
  <option value = "Програміст">56</option>
```

```
  <option value = "Системний адміністратор">44</option>
```

```
  <option value = "Маркетолог">42</option>
```

```
<br><input type="submit" value="Добавити">
```

```
</select>
```

```
<div>
```

```
<p>Відповідальний за ремонт</p>
```

```
<select name="number" autofocus>
```

```
<option>Олексій А.А.</option>
```

```
<option>Олександр К.С.</option>
```

```
</select>
```

```
Поштова скринька (Oleksandr.K@ua.ua)
```

```
</div>
```

```
</body>
```

```
</html>
```

HTML код сторінки генерації qr-коду

```

<!DOCTYPE html>
<html>
<head>
<title>QRCode test</title>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<meta name="viewport" content="width=device-width,initial-scale=1,user-
scalable=no" />
<script type="text/javascript" src="jquery.min.js"></script>
<script type="text/javascript" src="qrcode.js"></script>
</head>
<body>
<input id="text" type="text" value="anter html page" style="width:80%" /><br />
<div id="qrcode" style="width:200px; height:200px; margin-top:15px;"></div>

<script type="text/javascript">
var qrcode = new QRCode(document.getElementById("qrcode"), {
  width : 200,
  height : 200
});

function makeCode () {
  var elText = document.getElementById("text");

  if (!elText.value) {
    alert("Input a text");
    elText.focus();
    return;
  }
}

```

```

qrCode.makeCode(elText.value);
}

```

```

makeCode();

```

```

$("#text").
  on("blur", function () {
    makeCode();
  }).
  on("keydown", function (e) {
    if (e.keyCode == 13) {
      makeCode();
    }
  });
</script>
</body>

```

var QRCode

```

(function () {

  function QR8bitByte(data) {
    this.mode = QRMode.MODE_8BIT_BYTE;
    this.data = data;
    this.parsedData = [];

    // Added to support UTF-8 Characters
    for (var i = 0, l = this.data.length; i < l; i++) {

```

```

var byteArray = [];
var code = this.data.charCodeAt(i);

if (code > 0x10000) {
    byteArray[0] = 0xF0 | ((code & 0x1C0000) >>> 18);
    byteArray[1] = 0x80 | ((code & 0x3F000) >>> 12);
    byteArray[2] = 0x80 | ((code & 0xFC0) >>> 6);
    byteArray[3] = 0x80 | (code & 0x3F);
} else if (code > 0x800) {
    byteArray[0] = 0xE0 | ((code & 0xF000) >>> 12);
    byteArray[1] = 0x80 | ((code & 0xFC0) >>> 6);
    byteArray[2] = 0x80 | (code & 0x3F);
} else if (code > 0x80) {
    byteArray[0] = 0xC0 | ((code & 0x7C0) >>> 6);
    byteArray[1] = 0x80 | (code & 0x3F);
} else {
    byteArray[0] = code;
}

this.parsedData.push(byteArray);
}

this.parsedData = Array.prototype.concat.apply([], this.parsedData);

if (this.parsedData.length != this.data.length) {
    this.parsedData.unshift(191);
    this.parsedData.unshift(187);
    this.parsedData.unshift(239);
}

```

```
}
```

```
QR8bitByte.prototype = {
  getLength: function (buffer) {
    return this.parsedData.length;
  },
  write: function (buffer) {
    for (var i = 0, l = this.parsedData.length; i < l; i++) {
      buffer.put(this.parsedData[i], 8);
    }
  }
};
```

```
function QRCodeModel(typeNumber, errorCorrectLevel) {
  this.typeNumber = typeNumber;
  this.errorCorrectLevel = errorCorrectLevel;
  this.modules = null;
  this.moduleCount = 0;
  this.dataCache = null;
  this.dataList = [];
}
```

```
QRCodeModel.prototype={
  addData:function(data){
    var newData=new QR8bitByte(data);
    this.dataList.push(newData);
    this.dataCache=null;
  },
  isDark:function(row,col){
    if(row<0||this.moduleCount<=row||col<0||this.moduleCount<=col){
      throw new Error(row+","+col);
    }
  },
  return this.modules[row][col];
},
  getModuleCount:function(){
    return this.moduleCount;
  },
  make:function(){
    this.makeImpl(false,this.getBestMaskPattern());
  },
  makeImpl:function(test,maskPattern){
    this.moduleCount=this.typeNumber*4+17;
    this.modules=new Array(this.moduleCount);
    for(var row=0;row<this.moduleCount;row++){
      this.modules[row]=new
```

```
Array(this.moduleCount);for(var
col=0;col<this.moduleCount;col++){this.modules[row][col]=null;}}
```

```
this.setupPositionProbePattern(0,0);this.setupPositionProbePattern(this.moduleCount-
7,0);this.setupPositionProbePattern(0,this.moduleCount-
7);this.setupPositionAdjustPattern();this.setupTimingPattern();this.setupTypeInfo(test,
maskPattern);if(this.typeNumber>=7){this.setupTypeNumber(test);}
```

```
if(this.dataCache==null){this.dataCache=QRCodeModel.createData(this.typeNumber,
this.errorCorrectLevel,this.dataList);}
```

```
this.mapData(this.dataCache,maskPattern);},setupPositionProbePattern:function(row,c
ol){for(var r=-1;r<=7;r++){if(row+r<=-1||this.moduleCount<=row+r)continue;for(var
c=-1;c<=7;c++){if(col+c<=-
1||this.moduleCount<=col+c)continue;if(((0<=r&&r<=6&&(c==0||c==6))||((0<=c&&c<
=6&&(r==0||r==6))||((2<=r&&r<=4&&2<=c&&c<=4)))){this.modules[row+r][col+c]=t
rue;}else{this.modules[row+r][col+c]=false;}}},getBestMaskPattern:function(){var
minLostPoint=0;var pattern=0;for(var i=0;i<8;i++){this.makeImpl(true,i);var
lostPoint=QRUtil.getLostPoint(this);if(i==0||minLostPoint>lostPoint){minLostPoint=l
ostPoint;pattern=i;}}
```

```
return pattern;},createMovieClip:function(target_mc,instance_name,depth){var
qr_mc=target_mc.createEmptyMovieClip(instance_name,depth);var
cs=1;this.make();for(var row=0;row<this.modules.length;row++){var
y=row*cs;for(var col=0;col<this.modules[row].length;col++){var x=col*cs;var
dark=this.modules[row][col];if(dark){qr_mc.beginFill(0,100);qr_mc.moveTo(x,y);qr_
mc.lineTo(x+cs,y);qr_mc.lineTo(x+cs,y+cs);qr_mc.lineTo(x,y+cs);qr_mc.endFill();}}
```

```
return qr_mc;},setupTimingPattern:function(){for(var r=8;r<this.moduleCount-
8;r++){if(this.modules[r][6]!=null){continue;}
```

```
this.modules[r][6]=(r%2==0);}
```

```
for(var c=8;c<this.moduleCount-8;c++){if(this.modules[6][c]!=null){continue;}
```

```
this.modules[6][c]=(c%2==0);}},setupPositionAdjustPattern:function(){var
pos=QRUtil.getPatternPosition(this.typeNumber);for(var
i=0;i<pos.length;i++){for(var j=0;j<pos.length;j++){var row=pos[i];var
col=pos[j];if(this.modules[row][col]!=null){continue;}
```

```
for(var r=-2;r<=2;r++){for(var c=-2;c<=2;c++){if(r==2||r==2||c==
-2||c==2||((r==0&&c==0))){this.modules[row+r][col+c]=true;}else{this.modules[row+r]
[col+c]=false;}}}},setupTypeNumber:function(test){var
bits=QRUtil.getBCHTypeNumber(this.typeNumber);for(var i=0;i<18;i++){var
```

```

mod=(!test&&((bits>>i)&1)==1);this.modules[Math.floor(i/3)][i%3+this.moduleCount-8-3]=mod;}

for(var i=0;i<18;i++){var
mod=(!test&&((bits>>i)&1)==1);this.modules[i%3+this.moduleCount-8-3][Math.floor(i/3)]=mod;}},setupTypeInfo:function(test,maskPattern){var
data=(this.errorCorrectLevel<<3)|maskPattern;var
bits=QRUtil.getBCHTypeInfo(data);for(var i=0;i<15;i++){var
mod=(!test&&((bits>>i)&1)==1);if(i<6){this.modules[i][8]=mod;}else
if(i<8){this.modules[i+1][8]=mod;}else{this.modules[this.moduleCount-15+i][8]=mod;}}

for(var i=0;i<15;i++){var
mod=(!test&&((bits>>i)&1)==1);if(i<8){this.modules[8][this.moduleCount-i-1]=mod;}else
if(i<9){this.modules[8][15-i-1+1]=mod;}else{this.modules[8][15-i-1]=mod;}}

this.modules[this.moduleCount-8][8]=(!test);},mapData:function(data,maskPattern){var inc=-1;var
row=this.moduleCount-1;var bitIndex=7;var byteIndex=0;for(var
col=this.moduleCount-1;col>0;col-=2){if(col==6)col--;while(true){for(var
c=0;c<2;c++){if(this.modules[row][col-c]==null){var
dark=false;if(byteIndex<data.length){dark=(((data[byteIndex]>>>bitIndex)&1)==1);}

var mask=QRUtil.getMask(maskPattern,row,col-c);if(mask){dark=!dark;}

this.modules[row][col-c]=dark;bitIndex--;if(bitIndex===-1){byteIndex++;bitIndex=7;}}}

row+=inc;if(row<0||this.moduleCount<=row){row-=inc;inc=-inc;break;}}}}};QRCodeModel.PAD0=0xEC;QRCodeModel.PAD1=0x11;QRCodeModel.createData=function(typeNumber,errorCorrectLevel,dataList){var
rsBlocks=QRRSBlock.getRSBlocks(typeNumber,errorCorrectLevel);var buffer=new
QRBitBuffer();for(var i=0;i<dataList.length;i++){var
data=dataList[i];buffer.put(data.mode,4);buffer.put(data.getLength(),QRUtil.getLengthInBits(data.mode,typeNumber));data.write(buffer);}

var totalDataCount=0;for(var
i=0;i<rsBlocks.length;i++){totalDataCount+=rsBlocks[i].dataCount;}

if(buffer.getLengthInBits(>totalDataCount*8){throw new Error("code length overflow. ("
+buffer.getLengthInBits()
+ ">"
+totalDataCount*8

```

```

+"))");}
if(buffer.getLengthInBits()+4<=totalDataCount*8){buffer.put(0,4);}
while(buffer.getLengthInBits()%8!=0){buffer.putBit(false);}
while(true){if(buffer.getLengthInBits())>=totalDataCount*8){break;}

buffer.put(QRCodeModel.PAD0,8);if(buffer.getLengthInBits())>=totalDataCount*8){
break;}

buffer.put(QRCodeModel.PAD1,8);}

return
QRCodeModel.createBytes(buffer,rsBlocks);};QRCodeModel.createBytes=function(b
uffer,rsBlocks){var offset=0;var maxDcCount=0;var maxEcCount=0;var dcdata=new
Array(rsBlocks.length);var ecdata=new Array(rsBlocks.length);for(var
r=0;r<rsBlocks.length;r++){var dcCount=rsBlocks[r].dataCount;var
ecCount=rsBlocks[r].totalCount-
dcCount;maxDcCount=Math.max(maxDcCount,dcCount);maxEcCount=Math.max(m
axEcCount,ecCount);dcdata[r]=new Array(dcCount);for(var
i=0;i<dcdata[r].length;i++){dcdata[r][i]=0xff&buffer.buffer[i+offset];}

offset+=dcCount;var rsPoly=QRUtil.getErrorCorrectPolynomial(ecCount);var
rawPoly=new QRPolynomial(dcdata[r],rsPoly.getLength()-1);var
modPoly=rawPoly.mod(rsPoly);ecdata[r]=new Array(rsPoly.getLength()-1);for(var
i=0;i<ecdata[r].length;i++){var modIndex=i+modPoly.getLength()-
ecdata[r].length;ecdata[r][i]=(modIndex>=0)?modPoly.get(modIndex):0;}}

var totalCodeCount=0;for(var
i=0;i<rsBlocks.length;i++){totalCodeCount+=rsBlocks[i].totalCount;}

var data=new Array(totalCodeCount);var index=0;for(var
i=0;i<maxDcCount;i++){for(var
r=0;r<rsBlocks.length;r++){if(i<dcdata[r].length){data[index++]=dcdata[r][i];}}}

for(var i=0;i<maxEcCount;i++){for(var
r=0;r<rsBlocks.length;r++){if(i<ecdata[r].length){data[index++]=ecdata[r][i];}}}

return data;};var
QRMode={MODE_NUMBER:1<<0,MODE_ALPHA_NUM:1<<1,MODE_8BIT_B
YTE:1<<2,MODE_KANJI:1<<3};var QRErrorCorrectLevel={L:1,M:0,Q:3,H:2};var
QRMaskPattern={PATTERN000:0,PATTERN001:1,PATTERN010:2,PATTERN011
:3,PATTERN100:4,PATTERN101:5,PATTERN110:6,PATTERN111:7};var
QRUtil={PATTERN_POSITION_TABLE:[[],[6,18],[6,22],[6,26],[6,30],[6,34],[6,22,
38],[6,24,42],[6,26,46],[6,28,50],[6,30,54],[6,32,58],[6,34,62],[6,26,46,66],[6,26,48,7
0],[6,26,50,74],[6,30,54,78],[6,30,56,82],[6,30,58,86],[6,34,62,90],[6,28,50,72,94],[6,
26,50,74,98],[6,30,54,78,102],[6,28,54,80,106],[6,32,58,84,110],[6,30,58,86,114],[6,3

```

```

4,62,90,118],[6,26,50,74,98,122],[6,30,54,78,102,126],[6,26,52,78,104,130],[6,30,56,
82,108,134],[6,34,60,86,112,138],[6,30,58,86,114,142],[6,34,62,90,118,146],[6,30,54,
78,102,126,150],[6,24,50,76,102,128,154],[6,28,54,80,106,132,158],[6,32,58,84,110,1
36,162],[6,26,54,82,110,138,166],[6,30,58,86,114,142,170]],G15:(1<<10)|(1<<8)|(1<
<5)|(1<<4)|(1<<2)|(1<<1)|(1<<0),G18:(1<<12)|(1<<11)|(1<<10)|(1<<9)|(1<<8)|(1<<5
)|(1<<2)|(1<<0),G15_MASK:(1<<14)|(1<<12)|(1<<10)|(1<<4)|(1<<1),getBCHTypeIn
fo:function(data){ var d=data<<10;while(QRUtil.getBCHDigit(d)-
QRUtil.getBCHDigit(QRUtil.G15)>=0){d^=(QRUtil.G15<<(QRUtil.getBCHDigit(d)
-QRUtil.getBCHDigit(QRUtil.G15)));}

```

```

return((data<<10)|d)^QRUtil.G15_MASK;},getBCHTypeNumber:function(data){ var
d=data<<12;while(QRUtil.getBCHDigit(d)-
QRUtil.getBCHDigit(QRUtil.G18)>=0){d^=(QRUtil.G18<<(QRUtil.getBCHDigit(d)
-QRUtil.getBCHDigit(QRUtil.G18)));}

```

```

return(data<<12)|d;},getBCHDigit:function(data){ var
digit=0;while(data!=0){ digit++;data>>>=1;}

```

```

return digit;},getPatternPosition:function(typeNumber){ return
QRUtil.PATTERN_POSITION_TABLE[typeNumber-
1];},getMask:function(maskPattern,i,j){ switch(maskPattern){ case
QRMaskPattern.PATTERN000:return(i+j)%2==0;case
QRMaskPattern.PATTERN001:return i%2==0;case
QRMaskPattern.PATTERN010:return j%3==0;case
QRMaskPattern.PATTERN011:return(i+j)%3==0;case
QRMaskPattern.PATTERN100:return(Math.floor(i/2)+Math.floor(j/3))%2==0;case
QRMaskPattern.PATTERN101:return(i*j)%2+(i*j)%3==0;case
QRMaskPattern.PATTERN110:return((i*j)%2+(i*j)%3)%2==0;case
QRMaskPattern.PATTERN111:return((i*j)%3+(i+j)%2)%2==0;default:throw new
Error("bad
maskPattern:"+maskPattern);}},getErrorCorrectPolynomial:function(errorCorrectLen
gth){ var a=new QRPolynomial([1],0);for(var
i=0;i<errorCorrectLength;i++){ a=a.multiply(new
QRPolynomial([1,QRMath.gexp(i)],0));}

```

```

return
a;},getLengthInBits:function(mode,type){ if(1<=type&&type<10){ switch(mode){ case
QRMode.MODE_NUMBER:return 10;case QRMode.MODE_ALPHA_NUM:return
9;case QRMode.MODE_8BIT_BYTE:return 8;case QRMode.MODE_KANJI:return
8;default:throw new Error("mode:"+mode);}} else if(type<27){ switch(mode){ case
QRMode.MODE_NUMBER:return 12;case QRMode.MODE_ALPHA_NUM:return
11;case QRMode.MODE_8BIT_BYTE:return 16;case
QRMode.MODE_KANJI:return 10;default:throw new Error("mode:"+mode);}} else
if(type<41){ switch(mode){ case QRMode.MODE_NUMBER:return 14;case

```

```

QRMode.MODE_ALPHA_NUM:return 13;case
QRMode.MODE_8BIT_BYTE:return 16;case QRMode.MODE_KANJI:return
12;default:throw new Error("mode:"+mode);} }else{throw new
Error("type:"+type);} },getLostPoint:function(qrCode){ var
moduleCount=qrCode.getModuleCount();var lostPoint=0;for(var
row=0;row<moduleCount;row++){for(var col=0;col<moduleCount;col++){var
sameCount=0;var dark=qrCode.isDark(row,col);for(var r=-
1;r<=1;r++){if(row+r<0||moduleCount<=row+r){continue;}
for(var c=-1;c<=1;c++){if(col+c<0||moduleCount<=col+c){continue;}
if(r==0&&c==0){continue;}
if(dark==qrCode.isDark(row+r,col+c)){sameCount++;} } }
if(sameCount>5){lostPoint+=(3+sameCount-5);} } }
for(var row=0;row<moduleCount-1;row++){for(var col=0;col<moduleCount-
1;col++){var
count=0;if(qrCode.isDark(row,col))count++;if(qrCode.isDark(row+1,col))count++;if(
qrCode.isDark(row,col+1))count++;if(qrCode.isDark(row+1,col+1))count++;if(count
==0||count==4){lostPoint+=3;} } }
for(var row=0;row<moduleCount;row++){for(var col=0;col<moduleCount-
6;col++){if(qrCode.isDark(row,col)&&!qrCode.isDark(row,col+1)&&qrCode.isDark(
row,col+2)&&qrCode.isDark(row,col+3)&&qrCode.isDark(row,col+4)&&!qrCode.is
Dark(row,col+5)&&qrCode.isDark(row,col+6)){lostPoint+=40;} } }
for(var col=0;col<moduleCount;col++){for(var row=0;row<moduleCount-
6;row++){if(qrCode.isDark(row,col)&&!qrCode.isDark(row+1,col)&&qrCode.isDark
(row+2,col)&&qrCode.isDark(row+3,col)&&qrCode.isDark(row+4,col)&&!qrCode.i
sDark(row+5,col)&&qrCode.isDark(row+6,col)){lostPoint+=40;} } }
var darkCount=0;for(var col=0;col<moduleCount;col++){for(var
row=0;row<moduleCount;row++){if(qrCode.isDark(row,col)){darkCount++;} } }
var ratio=Math.abs(100*darkCount/moduleCount/moduleCount-
50)/5;lostPoint+=ratio*10;return lostPoint;} };var
QRMath={ glog:function(n){if(n<1){throw new Error("glog("+n+"")");}
return QRMath.LOG_TABLE[n];},gexp:function(n){while(n<0){n+=255;}
while(n>=256){n-=255;}
return QRMath.EXP_TABLE[n];},EXP_TABLE:new
Array(256),LOG_TABLE:new Array(256)};for(var
i=0;i<8;i++){QRMath.EXP_TABLE[i]=1<<i;}

```

```

for(var i=8;i<256;i++){QRMath.EXP_TABLE[i]=QRMath.EXP_TABLE[i-4]^QRMath.EXP_TABLE[i-5]^QRMath.EXP_TABLE[i-6]^QRMath.EXP_TABLE[i-8];}

for(var i=0;i<255;i++){QRMath.LOG_TABLE[QRMath.EXP_TABLE[i]]=i;}

function QRPolynomial(num,shift){if(num.length==undefined){throw new Error(num.length+"/"+shift);}

var offset=0;while(offset<num.length&&num[offset]==0){offset++;}

this.num=new Array(num.length-offset+shift);for(var i=0;i<num.length-offset;i++){this.num[i]=num[i+offset];}}

QRPolynomial.prototype={get:function(index){return this.num[index];},getLength:function(){return this.num.length;},multiply:function(e){var num=new Array(this.getLength()+e.getLength()-1);for(var i=0;i<this.getLength();i++){for(var j=0;j<e.getLength();j++){num[i+j]^=QRMath.gexp(QRMath.glog(this.get(i))+QRMath.glog(e.get(j)));}}

return new QRPolynomial(num,0);},mod:function(e){if(this.getLength()-e.getLength()<0){return this;}

var ratio=QRMath.glog(this.get(0))-QRMath.glog(e.get(0));var num=new Array(this.getLength());for(var i=0;i<this.getLength();i++){num[i]=this.get(i);}

```