جامعة تشرين
كلية الهندسة المعلوماتية
هندسة البرمجيات ونظم المعلومات
السنة الخامسة

مشروع هندسة النظم

إعداد:

الديم لقمان علوش ابراهيم عصام ابراهيم ابراهيم خضر دياب

إشراف:

م.مهام داؤود د.هبة أبو أحمد م.باسل خليل

السنة الدراسية 2024-2023

المتطلبات:

- React js: تم استخدامها لبناء واجهات المستخدم وتحقيق التفاعل معها عن طريق المكونات و ال jsx ومن معها استخدام :
  1. Mantine UI: توفر عناصر و تصاميم جاهزة للاستخدام مثل الجداول و استمارات البيانات و غيرها
  2. React router: للتوجه بين الصفحات
  3. React query و Axios: لجراء طلبات العميل للبيانات ومعالجتها
  4. Redux toolkit و React-redux: لدارة حالة التطبيق وسهولة تناقل البيانات بين المكونات في React
  5. React-hook-form: لتسهيل العامل مع استمارات البيانات وإرسال واستقبال
     البيانات من سيرفر ال node js

- Node js: بيئة تشغيل لل java script لتنفيذ الكواد الخاصة بال java script خارج المتصفح ومن معها استخدام
  1. Express js: إطار عمل لل node js لعجة تصميم ال api’s لتطبيقات الويب أكثر سهولة
  2. Cors: وسيط لمعالجة الطلبات والاستجابة ومنح سماحيات لportal للقيام بالعمليات المطلوبة

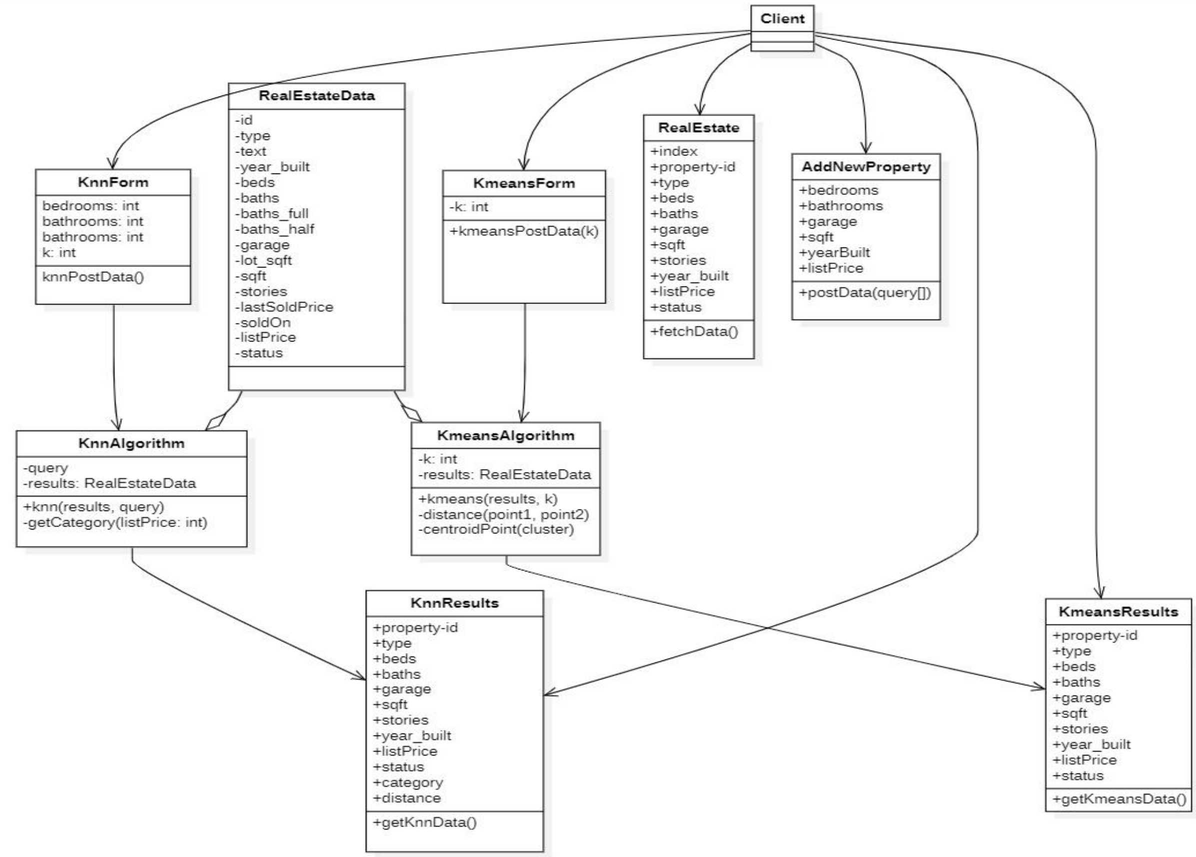
- Mysql: نظام إدارة قواعد بيانات علائقية من استخدامها مع xampp لتخزين البيانات والاستعلام عن البيانات
- Real Estate Chicago data set:

تعرض الداتا الست بيانات عن العقارات في مدينة شيكاغو تحتوي على 2000عنصر وتضم مواصفات مثل نوع العقار ومساحة العقار وسنة البناء وعدد الغرف والحمامات وسعر العقار الولي وسعر العقار عند آخر عملية بيع وتاريخ آخر عملية بيع وتاريخ بناء العقار يمكنك الطلع على التفاصيل على الرابط التالي:

[Real Estate Data Chicago 2024 (kaggle.com)]

المخططات:

مخطط ال class diagram:



الشكل)-1-(

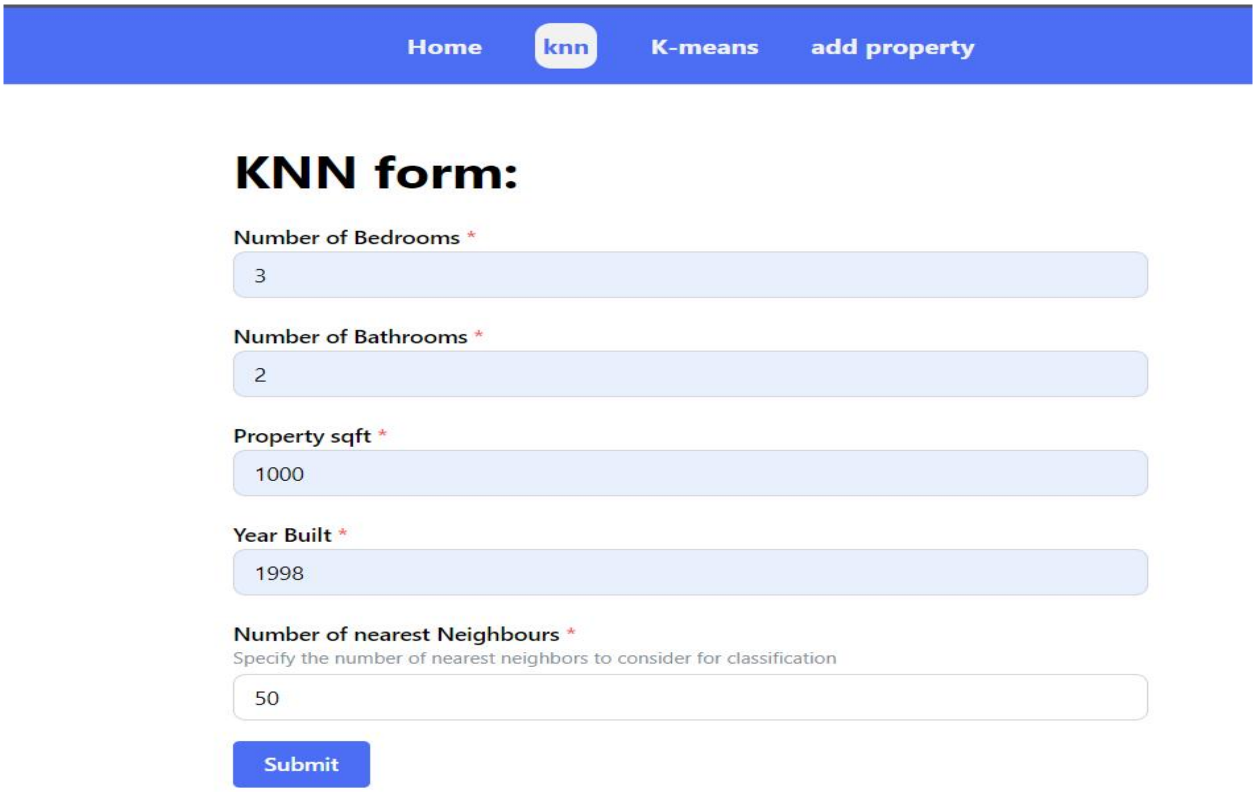
الخوارزميات المستخدمة في المشروع:

1- خوارزمية KNN:

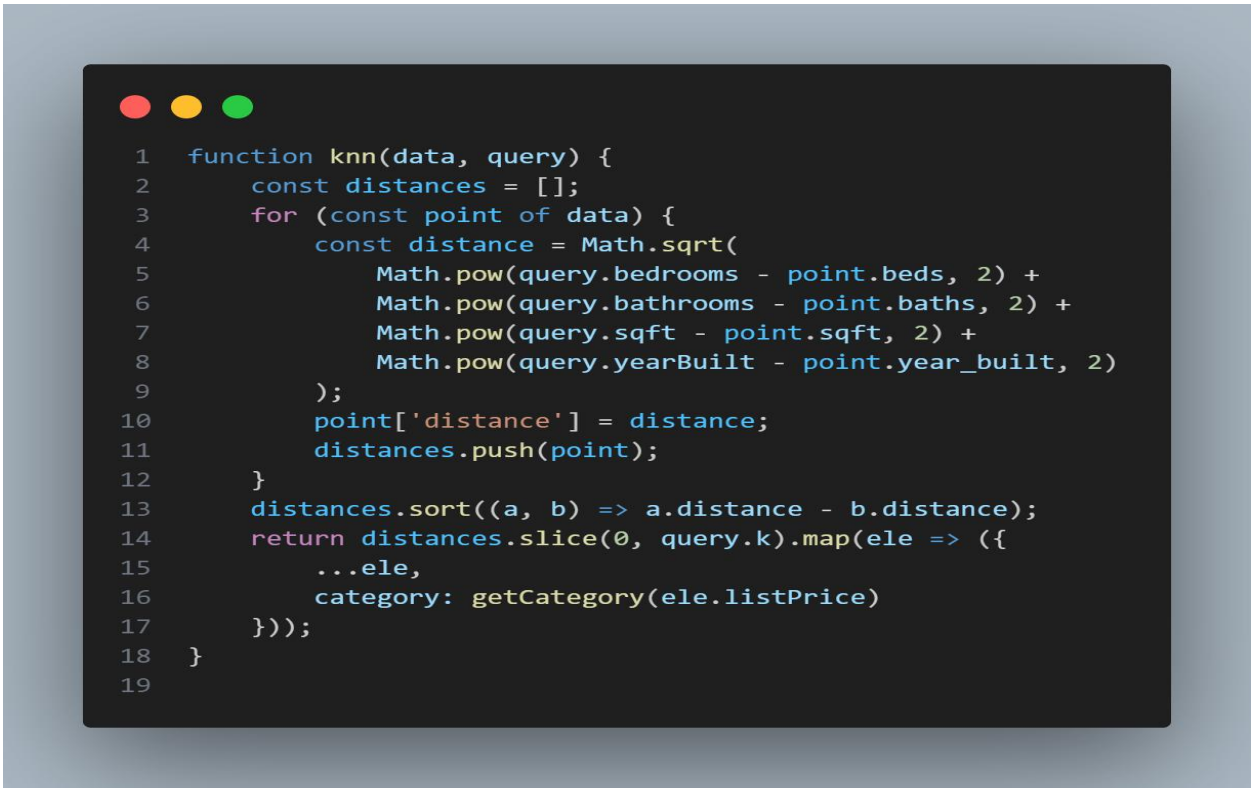
تم استخدام الخوارزمية للتعامل مع الدخل الموضح في الشكل)-2-(سيقوم التابع في الشكل)-3-(بحساب المسافات بين كل سطر مع البيانات من ال query ومن سيقوم بترتيب أول k عنصر من المصفوفة الناتجة وسيرسل سعر هذه العناصر الى التابع التصنيف الموضح في الشكل)-4-(والذي بدوره

سيقوم بتصنيف العقارات الى: cheap أو expensive أو affordable أو luxury

وسيكون الخرج عبارة عن جدول يحتوي العقارات مرتبة تصاعديا بناءً على قيمة distance و مصنفة كما هو موضح في الشكل (-5-)



الشكل (-2-)



```
function knn(data, query) {
    const distances = [];
    for (const point of data) {
        const distance = Math.sqrt(
            Math.pow(query.bedrooms - point.beds, 2) +
            Math.pow(query.bathrooms - point.baths, 2) +
            Math.pow(query.sqft - point.sqft, 2) +
            Math.pow(query.yearBuilt - point.year_built, 2)
        );
        point['distance'] = distance;
        distances.push(point);
    }
    distances.sort((a, b) => a.distance - b.distance);
    return distances.slice(0, query.k).map(ele => ({
        ...ele,
        category: getCategory(ele.listPrice)
    }));
}

```

الشكل (3-)

```
function getCategory(listPrice) {
    const cheapThreshold = 200000;
    const affordableThreshold = 400000;
    const expensiveThreshold = 600000;

    if (listPrice <= cheapThreshold) {
        return 'cheap';
    } else if (listPrice <= affordableThreshold) {
        return 'affordable';
    } else if (listPrice <= expensiveThreshold) {
        return 'expensive';
    } else {
        return 'luxury';
    }
}
```

الشكل (4-)

Home knn K-means add property

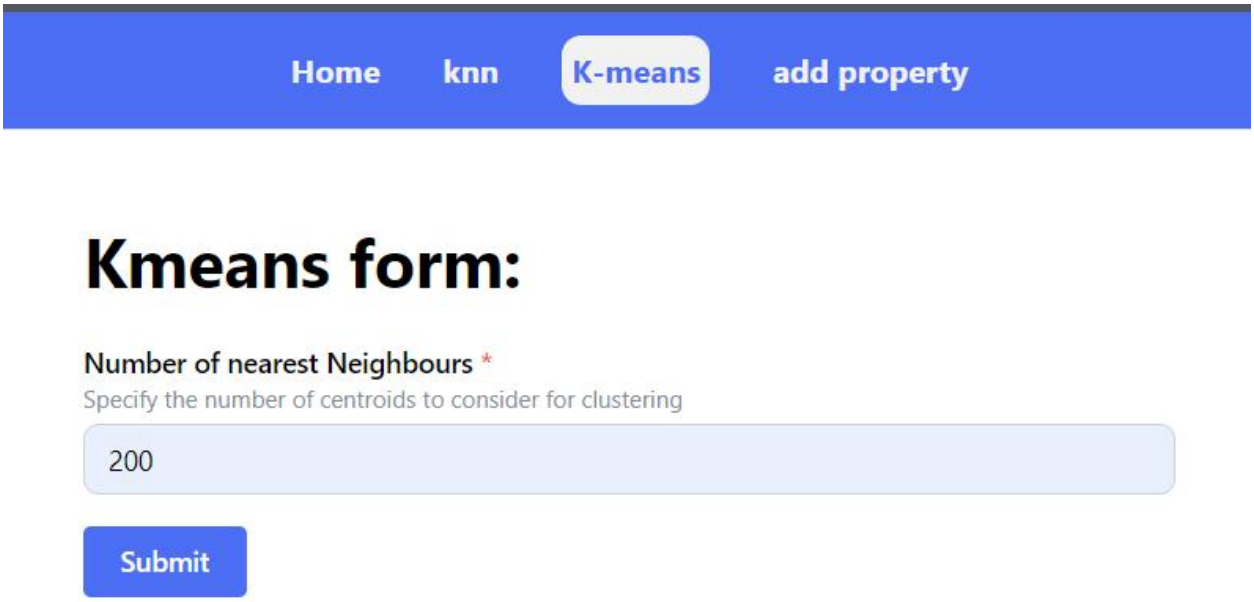
index	property-id	type	beds	baths	garage	sqft	stories	year_built	listPrice	status	category	distance
1	1072	condos	2	2	1	1000	3	1995	195000	for_sale	cheap	3.1622776601683795
2	645	land	2	0	2	1000	2	2002	29000	for_sale	cheap	4.58257569495584
3	827	condos	2	2	2	1000	3	2006	225000	for_sale	affordable	8.06225774829855
4	1913	townhomes	2	3	1	1000	2	1990	499900	for_sale	expensive	8.12403840463596
5	984	condos	1	1	1	1000	5	1989	199900	for_sale	cheap	9.273618495495704
6	1562	condos	2	1	2	1000	3	2008	225000	for_sale	affordable	10.099504938362077
7	642	land	2	0	2	1000	2	2009	19900	for_sale	cheap	11.224972160321824
8	643	land	2	0	2	1000	2	2009	25000	for_sale	cheap	11.224972160321824
9	644	land	2	0	2	1000	2	2009	60000	for_sale	cheap	11.224972160321824
10	21	multi_family	7	5	0	1000	1	2010	244400	for_sale	affordable	13
11	505	condos	1	2	1	1000	45	1981	450000	for_sale	expensive	17.11724276862369

الشكل (5-)

2- خوارزمية KMEANS:

تم استخدام الخوارزمية مع دخل عبارة عن عدد العناقيد المرادة الموضحة في الشكل)-6-(سيقوم النظام بتوليد الشكل)-7-(النقاط المركزية الولية بشكل عشوائي

ومن ثم تعيين العناقيد في الشكل)-8-(إلى أن يتم تقارب النقاط المركزية والخروج موضح بالشكل)-9-(



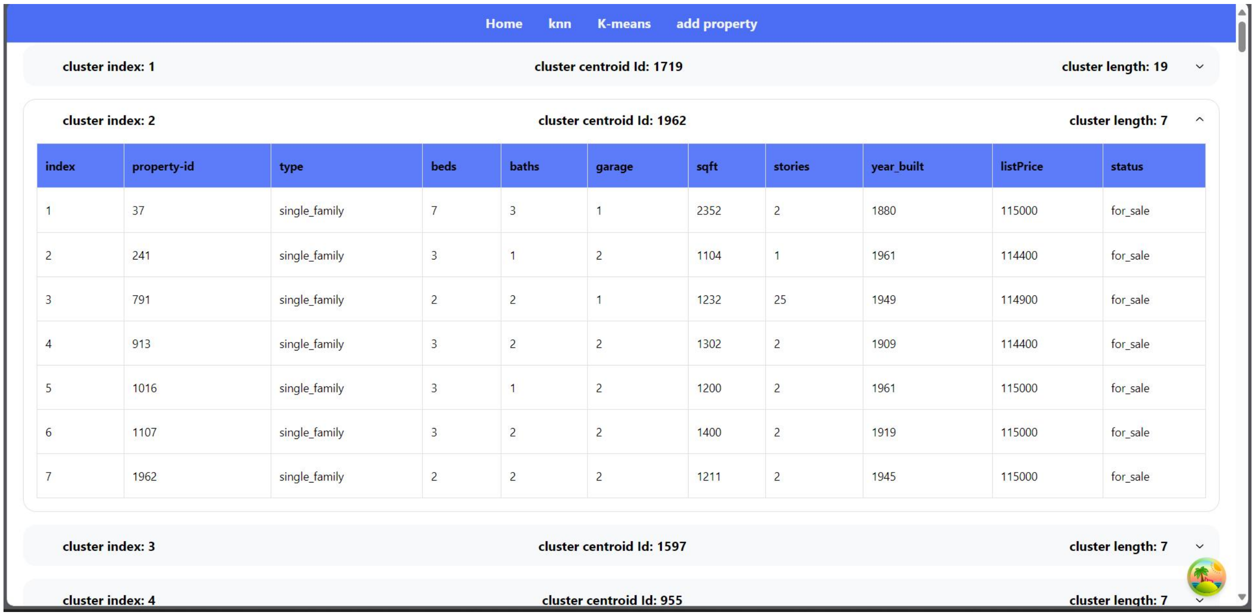
الشكل)-6-(

```
function kmeans(dataset, k) {
    const centroids = [];
    const centroidsIds = [];
    for (let i = 0; i < k; i++) {
        const index = Math.floor(Math.random() * dataset.length);
        centroidsIds.push(index + 1); // Use index directly
        centroids.push(dataset[index]);
    }

    let clusters = new Array(k).fill().map(() => []);
    let hasConverged = false;

    while (!hasConverged) {
        for (let i = 0; i < dataset.length; i++) {
            let minDist = Infinity;
            let clusterIndex = -1;
            for (let j = 0; j < k; j++) {
                const dist = distance(dataset[i], centroids[j]);
                if (dist < minDist) {
                    minDist = dist;
                    clusterIndex = j;
                }
            }
            clusters[clusterIndex].push(dataset[i]);
        }
        hasConverged = true;
        for (let i = 0; i < k; i++) {
            if (Array.isArray(centroids[i])) {
                const newCentroid = centroidPoint(clusters[i]);
                const isSameCentroid = centroids[i].every((val, index) => val === newCentroid[index]);
                if (!isSameCentroid) {
                    hasConverged = false;
                    centroids[i] = newCentroid;
                }
            }
        }
        if (!hasConverged) {
            clusters = new Array(k).fill().map(() => []);
        }
    }
    return {
        clusters, centroidsIds
    };
}
```

الشكل (7-)

```
function distance(point1, point2) {
    return Math.sqrt(
        Math.pow(point1.beds - point2.beds, 2) +
        Math.pow(point1.baths - point2.baths, 2) +
        Math.pow(point1.sqft - point2.sqft, 2) +
        Math.pow(point1.year_built - point2.year_built, 2) +
        Math.pow(point1.listPrice - point2.listPrice, 2)
    );
}


function centroidPoint(cluster) {
    const sum = { beds: 0, baths: 0, sqft: 0, year_built: 0, listPrice: 0 };
    for (let i = 0; i < cluster.length; i++) {
        sum.beds += cluster[i].beds;
        sum.baths += cluster[i].baths;
        sum.sqft += cluster[i].sqft;
        sum.year_built += cluster[i].year_built;
        sum.listPrice += cluster[i].listPrice;
    }
    for (const key in sum) {
        sum[key] /= cluster.length;
    }
    return sum;
}
```

الشكل (8-)



index	property-id	type	beds	baths	garage	sqft	stories	year_built	listPrice	status
1	37	single_family	7	3	1	2352	2	1880	115000	for_sale
2	241	single_family	3	1	2	1104	1	1961	114400	for_sale
3	791	single_family	2	2	1	1232	25	1949	114900	for_sale
4	913	single_family	3	2	2	1302	2	1909	114400	for_sale
5	1016	single_family	3	1	2	1200	2	1961	115000	for_sale
6	1107	single_family	3	2	2	1400	2	1919	115000	for_sale
7	1962	single_family	2	2	2	1211	2	1945	115000	for_sale

الشكل(-9-)

ملاحظات:

- يجب أن يحتوي الجهاز على نسخة node js18 وما فوق وعلى نسخة TypeScirpt أيضا
- لتشغيل التطبيق يجب أن يتم:

1- فتح ملف باسم realEstateDataSetProject على vscode وفتح ال terminal وكتابة التعليمة npm i لمرة واحدة فقط لتنزيل ال packages من npm run dev في كل مرة لتشغيل التطبيق في المتصفح على ال port:5173

2- فتح الملف باسم back end على vscode وفتح الterminal وكتابة التعليمة npm i لمرة واحدة لتنزيل ال packages من npm run start في كل مرة لتشغيل السيرفر

3- فتح xampp و تشغيل apache و mysql و عمل import لملف الcsv