



جامعة دمشق

كلية الهندسة الميكانيكية والكهربائية

قسم هندسة الحواسيب والأتمتة

تطویر نظام مساعد لتخطیط الرحل

السياحة

مشروع تخرج أعد لنيل درجة الإجازة في هندسة الحواسيب والأتمتة

إعداد الطلاب:

مصعب سمير ديوب

محمد حسين الأعمى

محمد مهند الكيالي

المشرف العلمي: د. كريستين زينية

العام الدراسي

2023-2024

كلمة شكر

نحمد الله عز وجل الذي وفقنا في إتمام هذا المشروع
والذي ألهمنا الصحة والعافية والعزيمة. فالحمد لله حمداً كثيراً.

نتقدم بجزيل الشكر والتقدير للدكتورة المهندسة كريستين زينية
على كل ما قدمته لنا من توجيهات ومعلومات قيمة
أملين أن تكون عند حسن ظنها

إلهراء

إلى من لا يغيب عن فكري وقلبي... إلى من لا رثاء يكفيه... إلى المعلم الأول وصاحب الفضل دائمًا
رحمه الله...
...

أبي الغالي
سمير ديوب

إلى من تعجز الكلمات عن شكرها... من يعجز القلب عن وصفها... من تقاسمت معي المتابع
دون كلل...
...

أمي الغالية
وفاء الباشا

إلى سndي في هذه الحياة... ومن شاركوني كل شيء... إلى من لن انسى فضلهم...
...

أخوتي مقداد، ميس، مي، لما، نغم

إلى من أعانتني لإكمال الطريق... ومن هونت على الصعاب... إلى من كان نجاحي هدفًا لها...
رؤى خيربك

صعب سمير ديوب

الفهرس

10	الملخص
11	الفصل الأول: الإطار النظري والدراسات السابقة
12	1-1 مِبيان (graph)
12	2-1 خوارزمية Dijkstra
14	3-1 أنظمة الاقتراح
15	1-3-1 التصفية التعاونية (Collaborative filtering)
15	2-3-1 التصفية استناداً إلى المحتوى (Content-based filtering)
15	3-3-1 أنظمة الاقتراح المختلطة (Hybrid recommender systems)
16	4-3-1 مقارنة بين خوارزميات أنظمة الاقتراح
17	4-1 المكتبات المستخدمة
17	1-4-1 مكتبة React
17	2-4-1 إطار عمل Laravel
18	5-1 MySQL
19	6-1 Dense layer
19	7-1 LSTM Layer
20	8-1 الدراسات السابقة
20	1-8-1 موقع Trip Advisor
20	2-8-1 مشروع Travel Recommendation System
21	3-8-1 موقع Lonely Planet

21	9-1- المقارنة مع الدراسات السابقة
23	الفصل الثاني: التحليل والتصميم
24	1-2- مرحلة التحليل
24	1-1-2- الهدف من المشروع
24	2-1-2- تحديد متطلبات المشروع
25	2-2- مرحلة التصميم
25	1-2-2- تصميم الخوارزمية
33	Flowchart diagrams -2-2-2
36	Site Map Diagram -3-2-2
37	4-2-2- مخطط حالة الاستخدام (Use Case Diagram)
38	5-2-2- EERD Diagram
39	الفصل الثالث: نظام الاقتراح
40	1-3- البيانات (dataset)
41	3-2- تحديد سمات نظام الاقتراح (Recommender system features)
42	3-3- المعالجة المسبيقة للبيانات
42	4-3- تقسيم البيانات لتدريب وختبار
42	5-3- بنية الشبكة العصبية
42	1-5-3- خوارزمية content-based filtering والشبكات العصبية
44	2-5-3- التوقع (regression) والشبكات العصبية (Dense)
44	3-5-3- التوقع (regression) والشبكات العصبية (LSTM)

45	4-5-3 - المقارنة بين الطرق الثلاث
46	الفصل الرابع: التطبيق العملي والنتائج
47	1-4 - جمع البيانات
48	2-4 - الواجهات الخلفية (Back-end)
48	1-2-4 - اللغات المستخدمة
49	2-2-4 - إطار عمل Laravel
54	3-2-4 - البرامج المستخدمة
56	3-4 - اللغات والمكتبات المستخدمة في بناء النماذج (models)
57	4-4 - الواجهة الامامية (Front-end)
57	1-4-4 - اللغات المستخدم
58	2-4-4 - المكتبات المستخدمة
59	3-4-4 - واجهات الموقع
65	4-4-4 - واجهات المدير (Admin UIs)
67	1-5 - التطبيق والنتائج:
67	1-5-5 - مثال (1)
72	2-5-5 - مثال (2)
76	3-5-5 - مثال (3)
81	الخاتمة
82	المراجع

الشكل 1-1: المخطط التدفقي لخوارزمية dijkstra	13
الشكل 1-2: معمارية خلية LSTM	19
الشكل 2-1: المخطط التدفقي لتحديد انواع الأماكن	28
الشكل 2-2: المخطط التدفقي لتصنيص ميزانية لكل نوع مكان	29
الشكل 2-3: المخطط التدفقي للحالة الاقتصادية	32
الشكل 2-4: المخطط التدفقي لتوليد الرحلة	33
الشكل 2-5: المخطط التدفقي لجلب البيانات	34
الشكل 2-6: المخطط التدفقي لبناء المبيان	35
الشكل 2-7: site map diagram	36
الشكل 2-8: مخطط حالة الاستخدام	37
الشكل 2-9: EERD Diagram	38
الشكل 3-1: المخطط الصندوقي لنظام الاقتراح	41
الشكل 4-1: دورة حياة الطلب-الاستجابة	52
الشكل 4-2: برنامج XAMPP	55
الشكل 4-3: برنامج postman	55
الشكل 4-5: صفحة الرحلة	59
الشكل 4-6: صفحة تصفح الرجل	59
الشكل 4-7: صفحة الملف الشخصي	59
الشكل 4-8: صفحة حولنا	59
الشكل 4-9: صفحة البحث عن اماكن	59
الشكل 4-10: الصفحة الرئيسية	59
الشكل 4-11: صفحة انشاء حساب	60
الشكل 4-12: صفحة تسجيل الدخول	60
الشكل 4-13: واجهة التعامل مع الرحلة	60

الجدول 1-1: مقارنة بين خوارزميات أنظمة الاقتراح.....16

الجدول 3-1: المقارنة بين النماذج.....45

الملخص

نظراً لزيادة السياحة وصعوبة التخطيط للرحلات السياحية تم تطوير النظام لتسهيل تخطيط الرحلات السياحية وتحقيق تجربة سياحية مخصصة ومرنة. يستخدم النظام معلومات المستخدم، مثل الميزانية وأنواع الأماكن المفضلة والبلد المرغوب زيارته وعدة معلومات أخرى، ليولد له رحلة سياحية كاملة بناء على مدخلاته.

يتبع المشروع طريقة شاملة للوصول إلى الهدف. أولاً، يتم جمع معلومات المستخدم بشكل مفصل، بما في ذلك أنواع الأماكن المفضلة، الميزانية، الأطعمة المفضلة، تاريخ الرحلة، عدد الشخصين المسافرين، عدد أيام الرحلة والأماكن المحددة المرغوب زيارتها. ثانياً، يتم اقتراح عدة أماكن باستخدام الذكاء الصنعي ليختار المستخدم منها. بعد ذلك يتم معالجة المدخلات وتنفيذ عدة عمليات فلترة لإيجاد الأماكن المناسبة للمستخدم، ثم يتم توليد مسار يتصف بأنه الأقصر من حيث المسافة والأكثر ملائمة لميزانية المستخدم.

يتم تنظيم هذا المسار كجدول زمني شامل يحتوي على جميع أماكن الرحلة كالفنادق، المطاعم والأماكن الأثرية. يتم احتساب المسافة والوقت التقديرية المستغرق للانتقال بين الأماكن، ويتم عرض وسيلة النقل الموصى بها. أيضاً يقوم النظام بعرض حجوزات الطيران والفنادق الازمة للرحلة، حيث يكون للمستخدم كامل الحرية بتثبيت هذه الحجوزات أو رفضها.

يستخدم المشروع خوارزمية Dijkstra للحصول على أقصر مسار وخوارزمية content-based filtering لاقتراح الأماكن.

نتائج المشروع تتمثل في توفير تجربة سفر مخصصة ومرحة للمستخدم. يستعيد المستخدم من توصيات مباشرة تتوافق مع اهتماماته وتفضيلاته الشخصية، ويحصل على جدول زمني ومسار مفصلين لتجويهه خلال رحلته وضمان زيارة جميع الأماكن المهمة.

الفصل الأول: الإطار النظري والدراسات السابقة

يتضمن هذا الفصل التعريف الأساسية وكافة الأمور النظرية التي تم الاستفادة منها في بناء المشروع ومنها بنية المعطيات مبيان (graph data structure) خوارزمية Dijkstra's algorithm، وأنظمة الاقتراح (Recommender Systems) والمكتبات المستخدمة.

1-1- مبيان (graph):

هو هيكل بيانات غير خطى يتتألف من مجموعة من العقد (Nodes or Vertices) والحواف (Edges). العقد هي الوحدات الأساسية في الرسم البياني. الحواف هي خطوط تربط أي عقدتين في الرسم البياني. بشكل أكثر صياغة يتم تمثيل الرسم البياني بمجموعة من العقد (V) ومجموعة من الحواف (E). يتم التعبير عن الرسم البياني كما يلي $[[2 G(V,E)]$.

يتم بناء مبيان لكل يوم في الرحلة بناء على مدخلات المستخدم.

1-2- خوارزمية Dijkstra:

خوارزمية Dijkstra هي خوارزمية تستخدم في مجالات الرسم البياني وعلوم الحاسوب. تهدف هذه الخوارزمية إلى حساب أقصر مسار بين عقدتين في الرسم البياني المرجعي.

تعمل خوارزمية Dijkstra على الرسم البياني الموجه (directed graph)، ويمكن استخدامها لحساب أقصر مسار بين عقدة بداية وعقدة نهاية عن طريق حساب الأطوال الدنيا للمسارات المحتملة.

خطوات عمل خوارزمية Dijkstra:

1. تعيين العقدة البداية والتعليمات الأولى:

– تعيين عقدة البداية ووضعها في مجموعة العقد المفتوحة.

– تعيين التكالفة الأولية لجميع العقد المتصلة بعقدة البداية بقيمة لا نهائية.

– تعيين التكالفة الأولية لعقدة البداية بقيمة صفر.

2. حساب التكاليف الدنيا:

– اختيار العقدة ذات التكالفة الأقل من مجموعة العقد المفتوحة وتحديث التكاليف للعقد المتصلة بها.

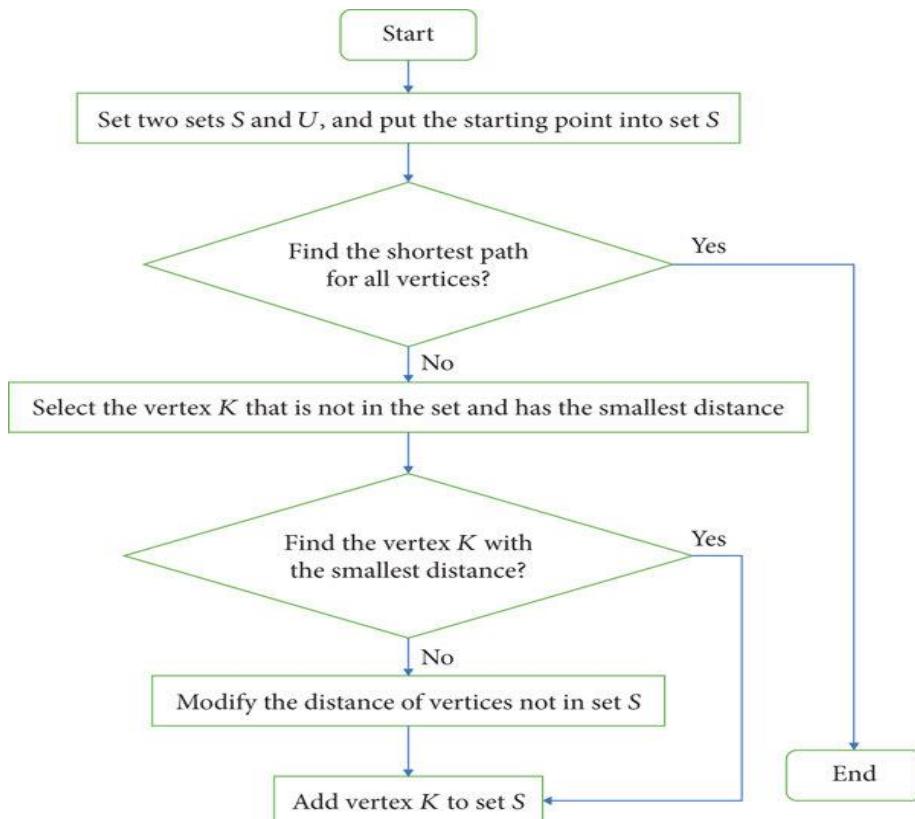
– إذا كانت التكاليف الجديدة أقل من التكاليف المحفوظة، فقم بتحديث التكاليف وتحديث العقدة السابقة للعقدة المحدثة.

– تكرار هذه الخطوة حتى يتم حساب التكاليف الدنيا لجميع العقد المتصلة بعقدة البداية.

3. العثور على أقصر مسار:

- بناء الطريق الأقصر من عقدة البداية إلى كل عقدة أخرى عن طريق اتباع العقد السابقة لكل عقدة.

يبين الشكل (1-1) المخطط التدفقي (flowchart) لخوارزمية Dijkstra [[4].



الشكل 1-1: المخطط التدفقي لخوارزمية Dijkstra

التعقيد الزمني (time complexity) لخوارزمية Dijkstra $O((V + E) \log V)$

التعقيد المكاني (space complexity) لخوارزمية Dijkstra $O(V)$

تم استخدام خوارزمية Dijkstra لإيجاد أقصر مسار في كل يوم من الرحلة.

1-3-أنظمة الاقتراح:

أنظمة الاقتراح أو التوصية هي أدوات وتقنيات برمجية تقدم اقتراحات للعناصر لتكون مفيدة للمستخدم. تتعلق الاقتراحات بمختلف عمليات صنع القرار، مثل العناصر التي يجب شراؤها، أو الموسيقى التي يجب الاستماع إليها، أو الأخبار التي يجب قرائتها.

"العنصر" هو المصطلح العام المستخدم للدلالة على ما يوصي به النظام للمستخدمين. يركز نظام التوصية عادة على نوع معين من العناصر (على سبيل المثال: الأفراص المدمجة أو الأخبار) وبالتالي تصميمه وواجهة المستخدم الخاصة به، وتقنية التوصية الأساسية المستخدمة لتوليد الاقتراحات جميعها تم تخصيصها لتكون مفيدة وفعالة لهذا النوع المحدد من العناصر.

تُقدم التوصيات على شكل قوائم مرتبة من العناصر. عند أداء هذا التصنيف، تحاول أنظمة الاقتراح التتبُّؤ بأي المنتجات أو الخدمات هي الأكثر ملائمة للمستخدم، استناداً إلى تفضيلات المستخدم والقيود المفروضة عليه. من أجل إكمال هذه المهمة الحسابية، تجمع أنظمة الاقتراح تفضيلات المستخدمين، والتي يتم التعبير عنها إما صراحةً، على سبيل المثال عن طريق تقديم تقييمات المنتجات، أو يتم استنتاجها من خلال تفسير أفعال المستخدم. على سبيل المثال، قد يعتبر نظام التوصية الانتقال إلى صفحة منتج معينة علامة ضمنية على تفضيل العناصر المعروضة على تلك الصفحة.

تعلم الآلة هو أحد التقنيات الرئيسية المستخدمة في تصميم وتنفيذ نظم التوصية. نظام التوصية هو أداة توصي بعناصر للمستخدمين استناداً إلى تفضيلاتهم وسلوكياتهم أو تشابههم مع مستخدمين آخرين. يتم استخدام خوارزميات التعلم الآلي لتحليل الأنماط وال العلاقات في البيانات لإجراء هذه التوصيات.

هناك عدة أنواع من خوارزميات تعلم الآلة المستخدمة عادةً في أنظمة الاقتراح، بما في ذلك:

◦ التصفية التعاونية (Collaborative filtering): خوارزمية تعتمد على التشابه بين المستخدمين وتفاعلاتهم السابقة مع العناصر.

◦ التصفية استناداً إلى المحتوى (Content-based filtering): خوارزمية توصية تعتمد على سمات العناصر وتفضيلات المستخدم.

◦ نظم التوصية المختلطة (Hybrid recommender systems): تركيب بين التصفية التعاونية والتصفية استناداً إلى المحتوى، والذي يؤدي في كثير من الأحيان إلى توصيات أكثر دقة.

يتم تدريب خوارزميات التعلم الآلي على مجموعة بيانات كبيرة لإنشاء التوصيات، ويمكن أن تحسن التوصيات مع مرور الوقت من خلال تحديث الخوارزميات باستخدام بيانات جديدة وتغذية ردود الفعل من المستخدمين [1].

1-3-1- التصفية التعاونية (Collaborative filtering):

الوصية القائمة على التصفية التعاونية هي نهج مستخدمة على نطاق واسع في مجال أنظمة الاقتراح. تهدف إلى تقديم توصيات مخصصة للمستخدمين بناء على سلوكياتهم السابقة وسلوكيات المستخدمين الآخرين. في التصفية التعاونية، يولد النظام توصيات من خلال تحديد المستخدمين المشابهين وإجراء التبؤات بناء على أوجه التشابه بينهم. يتم فيها اقتراح العناصر لمستخدم ما بناء على تقييمات المستخدمين اللذين أعطوا تقييمات مشابهة له [1].

1-3-2- التصفية استناداً إلى المحتوى (Content-based filtering):

التصفية المستندة إلى المحتوى هي نهج مستخدمة على نطاق واسع لتقديم التوصيات والتي يوصى فيها بالعناصر للمستخدمين بناء على تفضيلاتهم لمحتوى معين. يتم وصف العناصر باستخدام مجموعة من الميزات أو السمات (features)، مثل النوع أو المخرج أو الممثل، ويتم إنشاء التوصيات استناداً إلى التشابه بين هذه الميزات وتفضيلات المستخدم السابقة. تم استخدام هذه الطريقة على نطاق واسع في مختلف المجالات، وقد ثبت أنها فعالة في تقديم توصيات مخصصة للمستخدمين. يتم فيها اقتراح عنصر لمستخدم ما بناء على ميزات المستخدم وميزات العنصر [1].

1-3-3- أنظمة الاقتراح المختلطة (Hybrid recommender systems):

يجمع نظام التوصية المختلط بين خوارزميات توصية متعددة لتقديم توصية أكثر تخصيصاً ودقة للمستخدمين. يمكن أن يتغلب استخدام خوارزميات متعددة على قيود خوارزمية واحدة، على سبيل المثال الاعتماد المفرط على نوع واحد من المعلومات أو عدم القدرة على التعامل مع أنواع متعددة من البيانات. من خلال دمج الخوارزميات ومصادر البيانات المختلفة، يمكن أن يوفر نظام التوصية المختلط توصية أكثر شمولاً ودقة للمستخدمين، مما يؤدي إلى تجربة مستخدم أفضل وتحسين رضا المستخدم [1].

4-3-1 مقارنة بين خوارزميات أنظمة الاقتراح:

يوضح الجدول (1-1) مقارنة بين خوارزميات أنظمة الاقتراح.

الجدول 1-1: مقارنة بين خوارزميات أنظمة الاقتراح

المحاسن	المساوئ	الخوارزمية
<p>يمكن تقديم توصيات للمستخدمين الذين لديهم تفضيلات فريدة أو محددة بشكل جيد.</p> <p>يمكن التعامل مع مجموعة متنوعة من العناصر.</p>	<p>يمكن أن تعاني من مشكلة "البداية الباردة"، حيث قد يكون من الصعب تقديم توصيات للمستخدمين الجدد ذوي البيانات المحدودة. يمكن أن يؤدي إلى تنوع محدود في التوصيات، حيث قد يوصي النظام فقط بالعناصر ذات السمات المشابهة ل تلك التي أعجب بها المستخدم في الماضي.</p>	Content-based filtering
<p>يمكن التعامل مع عدد كبير من المستخدمين والعناصر.</p> <p>يمكن أن تقدم توصيات للمستخدمين الذين لديهم تفضيلات أكثر عمومية.</p>	<p>يمكن أن يعاني من مشكلة "التاثير"، حيث قد يكون من الصعب العثور على مستخدمين مشابهين لعناصر أو مستخدمين معينين. يمكن أن تتأثر بسلوك عدد قليل من المستخدمين المؤثرين.</p>	Collaborative filtering
<p>يمكن التعامل مع مجموعة متنوعة من العناصر وعدد كبير من المستخدمين. يمكن أن تقدم توصية أكثر دقة، مع مراعاة كل من التفضيلات الشخصية للمستخدم وتفضيلات المستخدمين المشابهين.</p>	<p>يمكن أن يكون التصميم والتنفيذ أكثر تعقيدا مقارنة بالخوارزمية الواحدة. يمكن أن تكون أقل قابلية للتقسيم، حيث قد يكون من الصعب فهم عملية صنع القرار في النظام.</p>	Hybrid

بناء على هذه المقارنة قمنا باستخدام خوارزمية content-based filtering في نظام الاقتراح لأنها الخوارزمية التي تأخذ سمات المستخدمين وسمات العناصر بعين الاعتبار.

4-1- المكتبات المستخدمة:

1-4-1 - مكتبة React

React هي مكتبة JavaScript تبسيط عملية تطوير واجهات المستخدم التفاعلية (UIs). مع React، يمكن للمصممين إنشاء واجهات مستخدم بسيطة لكل حالة (state) في التطبيق، وستقوم المكتبة بتحديث وعرض المكونات المناسبة بفاءة كلما حدث تغيير في البيانات.

يمكن نموذج البرمجة التصريحية (declarative programming model) والمعمارية القائمة على المكونات (components) المطوريين من بناء واجهات مستخدم معقدة بسهولة.

توفر المكتبة مجموعة من الأدوات والميزات التي تسمح بتطوير سريع وفعال، بما في ذلك DOM (document object model) الافتراضي، الذي يقلل الحاجة إلى التلاعب المباشر بـ DOM المتصفح، ومجموعة غنية من المكونات القابلة لإعادة الاستخدام التي يمكن تخصيصها بسهولة لتناسب متطلبات التصميم المختلفة. بشكل عام، تقدم React طريقة سهلة وفعالة لإنشاء واجهات مستخدم تفاعلية يمكن صيانتها وتوسيعها بسهولة مع نمو التطبيق.

مميزات React:

- البرمجة المبسطة: React تقدم ما يُعرف بـ JSX والذي يجعل كتابة HTML داخل الكود أسهل بكثير. JSX يسمح للمطوريين بدمج ترميز يشبه HTML مباشرةً في كود JavaScript الخاص بهم، مما يجعله أكثر بديهية وإيجازًا.
- البرمجة التصريحية: React يتبع نموذج برمجة تصريحية. يمكن للمطوريين تغيير حالة المكونات، وReact يقوم تلقائيًا بتحديث DOM وفقًا لذلك. هذا يعني أنك لست بحاجة إلى التلاعب المباشر بـ DOM، مما يجعل تصميم واجهة المستخدم وتصحيح الأخطاء أكثر قابلية للتبؤ وأكثر سهولة.
- المعمارية المعتمدة على المكونات: يشجع React على معمارية مبنية على المكونات. يمكن للمطوريين إنشاء مكونات مخصصة وإعادة استخدامها في جميع أنحاء التطبيق. هذه البنية النمطية تجعل الكود أكثر قابلية للصيانة والمرونة، مما يحسن كفاءة التطوير بشكل عام [11].

2-4-1 - إطار عمل Laravel

Laravel هو إطار عمل قائم على لغة PHP ، مفتوح المصدر وقد اكتسب شعبية كبيرة منذ إنشائه في عام 2011. يجسد Laravel مبادئ الأناقة، البساطة، والقابلية للقراءة. كإطار عمل صديق للمطوريين، يقدم ميزات قوية ويتبع مبدئ

.Model-View-Controller (MVC) architectural pattern

بعض النقاط الرئيسية حول Laravel :

- إطار ويب يُستخدم Laravel بشكل أساسي لبناء تطبيقات الويب باستخدام PHP. يتولى العديد من المهام التي يمكن أن تكون معقدة لبناء من الصفر مثل التوجيه والمصادقة.
 - يتبع Laravel نمط معماري Model-View-Controller (MVC) وهذا الفصل بين الاهتمامات يسمح للمطوريين بتنظيم الكود بكفاءة أكبر ، مما يجعله أسهل في الصيانة والتوسع.
 - المجتمع والنظام البيئي: لدى Laravel مجتمع نشط ونظام بيئي واسع من الحزم الأولية التي توسيع وظائفه. يدعم تقنيات الواجهة الأمامية المختلفة، بما في ذلك React ، Vue ، Inertia ، وما يسمح للمطوريين ببناء تطبيقات كاملة Full-stack.
- باختصار ، Laravel هو إطار عمل PHP قوي يمكن المطوريين من إنشاء تطبيقات ويب عالية الجودة بكفاءة وأنقة. سواء كنت تبني مشروعًا صغيرًا أو تطبيقًا على نطاق واسع، يوفر Laravel الأدوات والهيكل اللازم للنجاح [7].

:MySQL-5-1

MySQL هو نظام إدارة قواعد بيانات علائقية (RDBMS) مفتوح المصدر. يستخدم لإنشاء وصيانة قاعدة بيانات ديناميكية للموقع الإلكترونية والتطبيقات. يتميز بسرعته، ومونته، وسهولة استخدامه، وهو مدعم من قبل شركة Oracle.

يدعم MySQL لغات البرمجة الشائعة مثل PHP ، و Java ، و Python ، و C++ ، مما يجعله متواافقًا مع العديد من التطبيقات والأنظمة. يستخدم في العديد من التطبيقات الكبيرة مثل WordPress ، و Facebook ، و Twitter. يتضمن MySQL ميزات مثل الاستعلامات المخزنة، والمشغلات، والتكرار، والتقسيم، والتحكم في الوصول المستند إلى الدور. يعتبر خيارًا ممتازًا للمشاريع التي تتطلب قاعدة بيانات قوية وموثوقة.

يتم تحديث MySQL بانتظام لتحسين الأداء وإضافة ميزات جديدة. يتميز مجتمع مستخدمين نشط يقدم الدعم والمساعدة.

باختصار ، MySQL هو نظام قواعد بيانات قوي ومرن يستخدم في مجموعة واسعة من التطبيقات، ويُعتبر العمود الفقري للعديد من المواقع والتطبيقات الإلكترونية حول العالم.

تم استخدام MySQL كنظام إدارة قواعد بيانات في المشروع.

:Dense layer-6-1

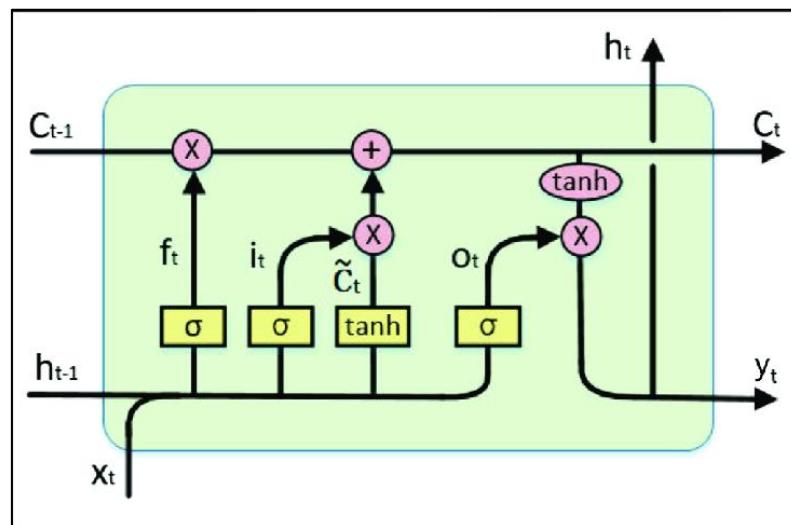
هي طبقة من طبقات الشبكة العصبية كاملاً الاتصال، مما يعني أن كل نeuron (نيرون) يتلقى مدخلات من جميع الخلايا العصبية في الطبقة السابقة. إن هذه الطبقة هي الأكثر استخداماً في بناء النماذج. في الخلفية، تقوم هذه الطبقة بضرب متوجه المصفوفة. القيمة المستخدمة في المصفوفة هي في الواقع قيم يمكن تدريبيها وتحديثها بمساعدة الانتشار العكسي (backpropagation). تم استخدام طبقة ال Dense لبناء النماذج.

:LSTM Layer-7-1

شبكة الذاكرة طويلة قصيرة-الامد أو LSTM، هي نوع من انواع الشبكة العصبية المتكررة (RNN) التي تكون فعالة جداً في التنبؤ بالسلسل الطويلة للبيانات مثل الجمل وأسعار الأسهم على مدى فترة زمنية. وهي تختلف عن شبكة التغذية الأمامية العادية نظراً لوجود حلقة تغذية خلفية في بنيتها. وتشمل أيضاً وحدة خاصة تعرف باسم خلية الذاكرة لاحتفاظ بالمعلومات السابقة لفترة أطول لإجراء تنبؤات فعالة. في الواقع، تعد LSTM مع خلية الذاكرة الخاصة بها نسخة محسنة من RNNs التقليدية التي لا يمكنها التنبؤ باستخدام مثل هذه السلسل الطويلة من البيانات وتواجه مشكلة التلاشي المتدرج (vanishing gradient) [9].

يبين الشكل (2-1) معمارية خلية LSTM [9].

تم استخدام طبقة ال LSTM لبناء إحدى النماذج.



الشكل 1-2: معمارية خلية LSTM

هذه هي الأمور النظرية التي تم الاستفادة منها في المشروع حيث انه تم استخدام خوارزمية Dijkstra لإيجاد أقصر مسار ممكن في كل يوم من الرحلة، وتم استخدام خوارزمية content-based filtering وطبقات LSTM وdense في تطوير نظام الاقتراح.

1-8-1- الدراسات السابقة:

هنا سيتم شرح الدراسات والتطبيقات المشابهة للنظام الذي تم تطويره.

1-8-1- موقع Trip Advisor

Tripadvisor هو موقع عبر الإنترن트 يوفر للمسافرين مجموعة واسعة من المعلومات حول الفنادق والمطاعم والأنشطة والأماكن السياحية في جميع أنحاء العالم. يعتمد على المشاركة النشطة للمستخدمين، الذين يشاركون تجاربهم من خلال المراجعات والآراء، وبالتالي يساهمون في إنشاء مجتمع عالمي من المسافرين. يمكن للمستخدمين البحث عن معلومات حول الوجهات، وقراءة التعليقات، والحصول على نصائح حول أماكن الإقامة والأنشطة المتوفرة. واحداً مميزات هذا الموقع (وهي ميزة لا تزال قيد التطوير) هي إمكانية توليد رحلة كاملة لمدينة محددة عن طريق الإجابة على عدة أسئلة ومن هنا انطلقت فكرة مشروعنا.

الأسئلة هي:

- 1- ما هي المدينة التي ت يريد زيارتها؟
- 2- ما هو تاريخ الرحلة وما هي مدتها (7 أيام كحد أقصى)؟
- 3- كيف ت يريد قضاء وقتك؟ (هنا يمكن للمستخدم الاختيار من ضمن عدة خيارات مثل: الطعام اللذيد، أهم المعالم السياحية الموجودة في الوجهة، الفن والمتحف، الخ...)

بعد إجابة المستخدم على هذه الأسئلة يقوم الموقع بتوسيع رحلة للمستخدم، كل يوم من هذه الرحلة يحتوي على مسار للأماكن السياحية والمطاعم التي يجب زيارتها، ويتم اقتراح فنادق ليبيت فيها المستخدم. يتيح للمستخدمين حجز الفنادق، الرحلات الجوية، والجولات السياحية عبر الإنترنط [5][6].

1-8-2- مشروع Travel Recommendation System by SIDDHI SHRIVASTAVA

يتضمن هذا المشروع بناء نظام اقتراح بالاعتماد على خوارزمية collaborative-filtering لاقتراح الأماكن السياحية في نيوزيلاند [8][9].

هذا المشروع هو عبارة عن تطبيق لخوارزمية collaborative-filtering باستخدام لغة python وباستخدام مكتبات TensorFlow وNumPy وpandas.

ت تكون ال dataset من 10,000 تقييم، 200 مستخدم و 400 مكان في نيوزيلاند. يبدأ المشروع باستعراض البيانات وخصائصها، ثم يتم اختيار الميزات، بعد ذلك يتم عمل إعادة معالجة لهذه الميزات، بعد ذلك يتم تقسيم البيانات لتدريب واختبار، وفي النهاية يتم تدريب النموذج. لا يحتوي هذا المشروع على واجهات او قواعد بيانات لأنه فقط عبارة عن بناء نظام اقتراح باستخدام python.

3-8- موقع Lonely Planet

موقع Lonely Planet هو أحد أشهر وأقدم الموقع الإلكتروني المخصصة للسفر والسياحة. تأسس الموقع عام 1973، وهو يقدم مجموعة واسعة من الموارد والمعلومات المفيدة للمسافرين.

يقدم الموقع معلومات شاملة حول الوجهات السياحية في جميع أنحاء العالم، بما في ذلك الأماكن التي يمكن زيارتها، وأفضل الأوقات للسفر، والأنشطة التي يمكن القيام بها.

يحتوي على مقالات وكتابات من قبل مسافرين محترفين وخبراء في السفر، تغطي مواضيع مثل نصائح السفر، والثقافة المحلية، والطعام، والتجارب الفريدة.

يحتوي الموقع على تقييمات ومراجعات للأماكن السياحية، الفنادق، المطاعم، والخدمات الأخرى، مما يساعد المسافرين على اتخاذ قرارات جيدة عند التخطيط لرحلتهم.

يتيح للمستخدمين حجز الفنادق، الرحلات الجوية، والجولات السياحية عبر الإنترنت.

Lonely Planet يعتبر مصدراً موثوقاً للعديد من المسافرين حول العالم لأنه من أقدم الموقع الإلكتروني على الانترنت التي تقدم معلومات للمسافرين [6][][].

9- المقارنة مع الدراسات السابقة:

بالاستناد إلى Trip advisor قمنا بتطوير نظام تخطيط رحل ولكن ليس على مدينة واحدة وإنما على بلد كامل بالإضافة لعدة أشياء أخرى غير موجودة في trip advisor مثل إنشاء رحلة مناسبة لميزانية المستخدم، اقتراح وسيلة المواصلات بين الأماكن.

يتفوق lonely planet على مشروعنا من حيث كمية المعلومات والمقالات التي يحتويها عن السفر وذلك لأنه من أقدم الموقع الموجودة على الانترنت، ولكن مشروعنا يتفوق على lonely planet بخاصية إنشاء رحلة سياحية واقتراح الأماكن للمستخدم باستخدام خوارزمية content-based filtering.

بالاستاد الى مشروع SIDDHI SHRIVASTAVA Travel Recommendation System تم الاستفادة من عمليات إعادة المعالجة و اختيار السمات و تصفية البيانات قبل عملية تدريب النماذج. يتتفوق مشروعنا على هذا المشروع لأن مشروعنا يأخذ بعين الاعتبار سمات خاصة بالمستخدم مثل عمره و جنسه بالإضافة الى تقييمات السابقة، بينما مشروع SIDDHI SHRIVASTAVA Travel Recommendation System فهو يأخذ فقط تقييمات المستخدمين السابقة بعين الاعتبار.

باستنادنا إلى هذه الدراسات السابقة يظهر بوضوح أن مشروعنا يقدم قيمة فريدة و ميزات إضافية للمستخدمين، لأنه لا يوجد حتى الان نظام يقوم بتنظيم رحلة المستخدم الى بلد كامل وبشكل متكامل و متناسب مع تفضيلاته و ميزانيته كما يقوم نظامنا.

الفصل الثاني: التحاليل والتصميم

2-1-مرحلة التحليل:

مرحلة التحليل هي الإجرائية التي يتم فيها تحديد ما يجب أن يقوم به نظام المعلومات، يتم في هذه المرحلة جمع وتحليل المتطلبات المتعلقة بالمشروع. هذه المرحلة تهدف إلى فهم متطلبات واحتياجات المشروع.

2-1-1- الهدف من المشروع:

مساعدة المستخدم على إنشاء رحل سياحية متكاملة عن طريق إجابة المستخدم على عدة أسئلة و اختياره لاماكن و تفضيلات محددة.

2-1-2- تحديد متطلبات المشروع:

يجب تحديد المتطلبات بشكل دقيق و محدد لتمكين التصميم والتطوير الفعال والسهل للنظام البرمجي. كلما تم تحديد المتطلبات بدقة أكبر، زادت سهولة التصميم والتطوير. تقسم المتطلبات في أي مشروع برمجي إلى متطلبات وظيفية ومتطلبات غير وظيفية.

المتطلبات الوظيفية:

1. تسجيل المستخدم: يجب أن يكون المستخدمون قادرون على إنشاء حسابات و تسجيل الدخول لهذه الحسابات للاستفادة من خدمات الموقع.
2. التعديل على الحساب: يجب أن يكون المستخدم قادر على رفع صورة شخصية وتغييرها و حذفها بالإضافة لتعديل كلمة المرور.
3. البحث عن أماكن: يجب أن يكون المستخدمون قادرون على البحث عن الأماكن السياحية واستعراض معلومات عنها.
4. توليد رحلة: توليد مسار رحلة كامل للمستخدم مع إمكانية التعديل على هذا المسار (إضافة مكان، حذف مكان).
5. خريطة تفاعلية: توفير خريطة تفاعلية لمسار رحلة المستخدم.
6. اقتراح أماكن: اقتراح أماكن للمستخدم باستخدام نظام اقتراح ليضمها في رحلته.

المتطلبات غير الوظيفية:

1. قابلية الاستخدام: واجهة المستخدم يجب أن تكون واضحة، وسهلة الاستخدام ويجب أن تكون متوافقة مع الشاشات المختلفة.
2. قابلية التوسيع: يجب أن يكون النظام قابل لإضافة وظائف جديدة، حيث يتم الأخذ بعين الاعتبار قابلية النظام للتوسيع والقدرة على إضافة الخدمات الجديدة دون إعادة العمل من البداية، وتشمل التوسيعية أيضا القدرة على التعديل على وظائف موجودة مسبقا.
3. الوثوقية: يجب أن يكون الموقع مستقرًا وموثوقًا، مع تقليل الأخطاء والانقطاعات التي تؤثر على تجربة المستخدم.
4. قابلية الصيانة: تسهيل عمليات الصيانة والإصلاح والتحديثات. تهدف قابلية الصيانة إلى تقليل التكلفة والجهد المطلوبين لإجراء الصيانة وتجنب تعطل النظام لفترات طويلة.

2-2-مرحلة التصميم:

تتضمن هذه المرحلة المعمارية الرئيسية لتصميم النظام وجميع الخطوات المتتبعة لتصميم الخوارزمية.

1-2-2- تصميم الخوارزمية:

يجب على المستخدم الإجابة على استبيان ليتم توليد مسار الرحلة بناء على إجاباته.

أسئلة الاستبيان:

- 1- ما هي المدينة التي تريد السفر منها؟
- 2- ما هي البلد الوجهة؟
- 3- ما هو تاريخ الرحلة؟
- 4- ما هو عدد الأشخاص المسافرين؟
- 5- تحديد عدد أيام الرحلة.
- 6- هل تهمك ميزانية الرحلة؟، إذا كان الجواب نعم يتم سؤال المستخدم سؤال آخر "هل تريد أرخص رحلة ممكنة؟" ، إذا كان الرد لا يجب على المستخدم تحديد ميزانية الرحلة.
- 7- تحديد أنواع الطعام المفضلة (تقليدي، بحري، طعام سريع، طعام فاخر، وحلويات).
- 8- تحديد أنواع الأماكن المفضلة (أماكن تسوق، أماكن ليلية، أماكن أثرية، وأماكن طبيعية).
- 9- تحديد مكان أو مجموعة أماكن محددة (فندق محدد، مكان أثري محدد).

بعد إجابة المستخدم على هذه الأسئلة يتم اتباع الخطوات التالية لتوليد الرحلة:

- يأتي الطلب إلى التابع `generate` المعرف في متحكم تم إنشاءه مسبقا يدعى `GenerateTripController` الذي يستقبل الطلب ك وسيط ويستخلص منه جميع المعلومات التي قام المستخدم بإدخالها.
- يتم التحقق من اتباع المعلومات المدخلة لعدة شروط منها:
 1. عدد الأيام المدخل من قبل المستخدم لا يتجاوز 30 يوم.
 2. الميزانية المدخلة لا تقل عن 500.
 3. أنواع الأماكن المفضلة لا تقل عن 2.
 4. عدد الأشخاص لا يتجاوز 30.
- 5. المستخدم قد قام بتسجيل دخول موجود في جدول المستخدمين في قاعدة البيانات.

- 6. المدينة المدخلة موجودة في قاعدة البيانات.
- 7. الدولة المدخلة موجودة في قاعدة البيانات.
- في حال عدم تحقق أي شرط من هذه الشروط تحدث مقاطعة ويتم إرسال رسالة خطأ ضمن الاستجابة تتضمن وصف لسبب الخطأ الذي حدث.
- بعد ذلك يتم تحديد عدد الأيام لكل مدينة في الرحلة عن طريق تابع معرف مسبقاً ضمن المتحكم يدعى: **citydays**
 - عدد الأيام يتم توزيعها وفق ما يلي:
$$\text{cityNumberOfDays} = \text{NumberOfDays} / \text{N_cities}$$

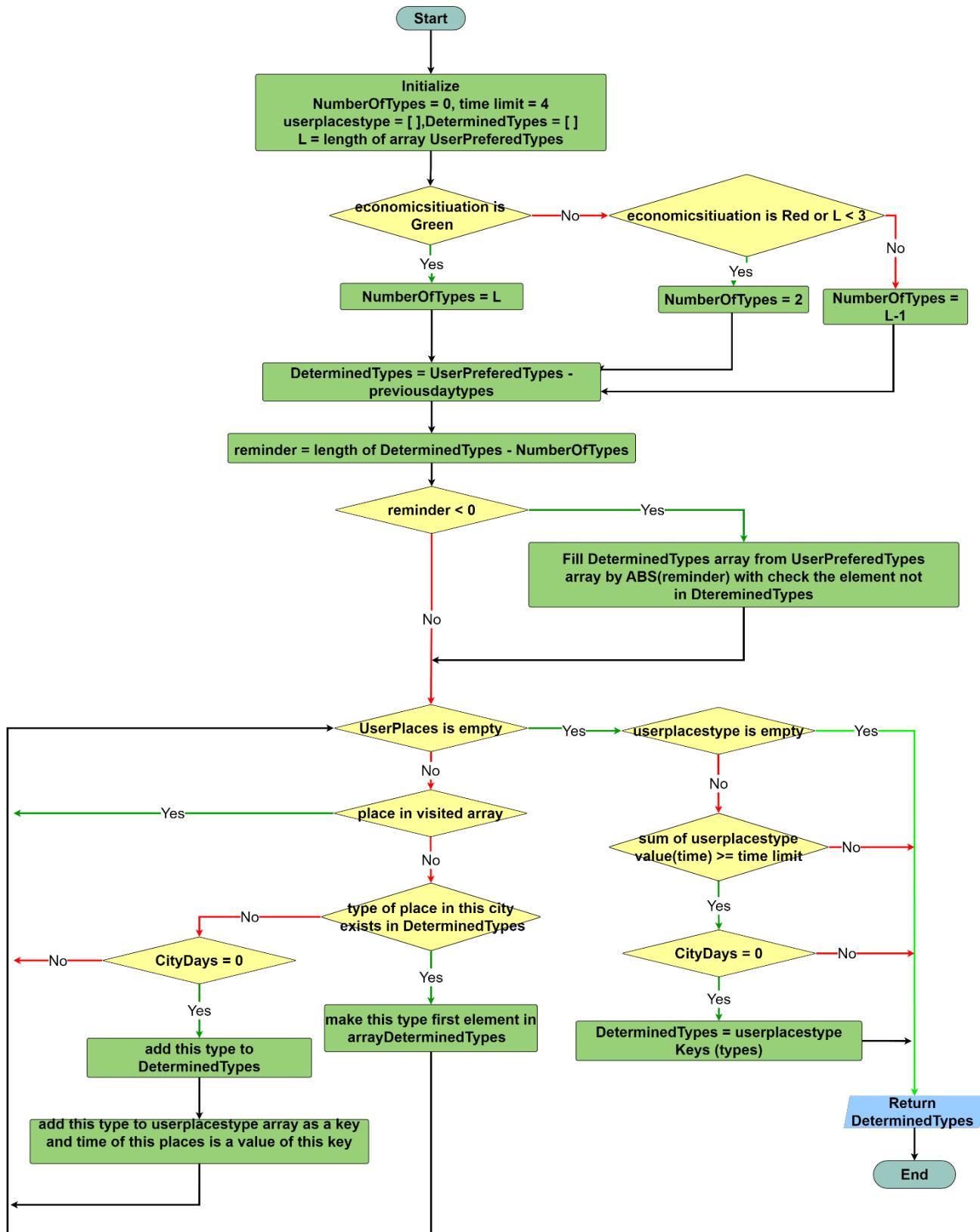
حيث **cityNumberOfDays** : عدد الأيام لكل مدينة، **NumberOfDays** هي عدد الأيام المدخلة، **N_cities** : عدد المدن.

 - أما الباقي من هذه القسمة فيتم توزيعه على مدن الدولة المدخلة التي تم جلبها من قاعدة البيانات وتخزينها في مصفوفة بدءاً من المدينة الأولى حتى ينتهي هذا الباقي.
 - يتم حساب كلف السفر بين المدن في الرحلة بما يتضمن كلفة السفر بين المدينة التي انطلق منها المستخدم وعاصمة الدولة التي يريد السفر إليها.
 - يتم اقتطاع كلف السفر بين المدن من الميزانية الكلية.
 - بعد ذلك من أجل كل يوم في الرحلة:
 - يتم تخصيص مبلغ لهذا اليوم اذ أن هذا المبلغ ليس ثابتاً لكل الأيام انما سيتغير بناء على عدة عوامل سراها لاحقاً في بقية الخطوات ويتم تقدير هذا المبلغ وفق:

$$\text{DayBudget} = (\text{Budget} - \text{Day cost}) / \text{NDays}$$

- حيث **DayBudget** : المبلغ المخصص لليوم، **Budget** : الميزانية المدخلة، **Day** : المبلغ المخصص لليوم
- كلفة اليوم السابق بقيمة ابتدائية معدومة، **NDays** : عدد الأيام المتبقية
- ان هذه الخطوة هي أحد الخطوات الرئيسية التي تساهم بشكل كبير في عدم تجاوز كلفة الرحلة للميزانية المدخلة من قبل الزيون ويتم تحديث المبلغ اليومي المخصص في كل يوم وفق المعادلة السابقة.

- بعد ذلك يتم تحديد انواع الاماكن التي سيتم زيارتها لهذا اليوم وذلك عن طريق استدعاء تابع معرف مسبقا في المتحكم .
- هذا التابع كما نرى في الشكل (2-1) الذي يبين المخطط التدفقي له يراعي اقل تكرار ممكن من حيث الانواع في ايام الرحلة كما يراعي عدد الاماكن التي سيتم زيارتها في اليوم وذلك بالاعتماد على متغير يتم تحديد قيمته في كل يوم تبعا للحالة الاقتصادية للرحلة والمعبر عنها عن طريق حساب الـ (Trip_balancing) كما سنرى لاحقا .
وفي حال اضافة المستخدم لأماكن يتم ضمان عدم تكرار نوع المكان الذي قام بإضافته ضمن الانواع التي تم اختيارها لهذا اليوم .

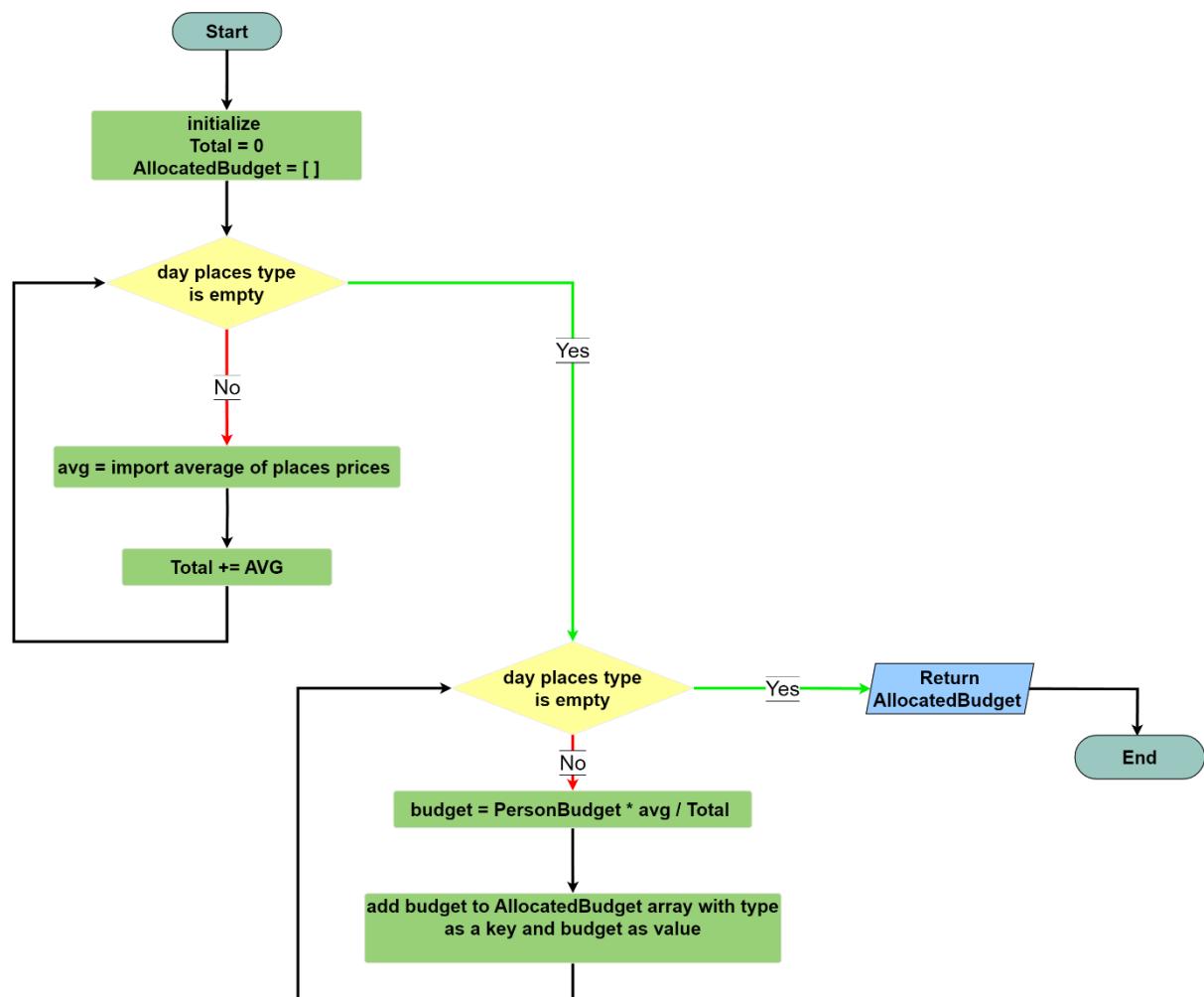


الشكل 2-1: المخطط التدفقى لتحديد انواع الأماكن

➢ يتم استدعاء تابع موجود في الصنف (DataImport) الذي يمثل أحد خدمات هذا المتحكم ووظيفته الرئيسية جلب الامكان من قاعدة البيانات وفق ما يلي:

1. تخصيص ميزانية لكل شخص لهذا اليوم وذلك لأن الاسعار الموجودة في قاعدة البيانات هي للشخص الواحد.

2. تخصيص ميزانية لكل مكان في هذا اليوم بما في ذلك المطاعم والفندق وذلك عن طريق استدعاء تابع يقوم هذا التابع كما نرى في الشكل (4) بتوزيع المبلغ المخصص لهذا اليوم على الانواع كنسب مئوية وذلك بالاعتماد على متوسط اسعار الاماكن لكل نوع في قاعدة البيانات ويتم حساب المبلغ المخصص لكل نوع من الاماكن.



الشكل 2-2: المخطط التدفقي لـ تخصيص ميزانية لكل نوع مكان

3. وبنفس الطريقة يتم تخصيص وقت لكل نوع من الاماكن باستثناء المطاعم والفندق لأن الوقت المخصص لها ثابت ساعة للمطعم وساعتين للفندق.

4. يتم جلب الاماكن من قاعدة البيانات بعد تصفيتها لتناسب الانواع التي اختارها المستخدم ويتم تصفيتها ايضا بحيث يتم جلب الاماكن الاقرب للسعر والوقت اللذان تم حسابهما في الخطوتين 3 و 4 علما انه بعد جلب اماكن كل نوع سيتم اقتطاع كلفة هذا النوع من الميزانية المخصصة لليوم وبعد ذلك سيتم استدعاء تابعي الخطوة 3 و 4 في كل مرة وذلك من أجل مراعاة الدقة في الوصول الى اقرب رقم ممكن للميزانية المدخلة من قبل المستخدم.

5. الان لدينا مجموعة من الاماكن لكل نوع من الانواع المخطط لزيارتها لهذا اليوم.

► يتم تمثيل هذه الاماكن عن طريق مبيان(graph) وذلك من خلال التابع المخصص لبناء المبيان والمعرف في الصنف (custumgraph) الذي يمثل أحد الخدمات التابعة للمتحكم في هذا الصنف يتم الاستعانة بالمكتبة (GenerateTripController) لبناء رسم بياني للأماكن وفق ما يلي:

- انشاء مستوى لكل نوع من الاماكن على ان عقدة الانطلاق سواء كان المطار او الفندق. وذلك تبعا لعدة امور منها:
 - هل هذا اليوم هو اليوم الأول من الايام المخصصة لهذه المدينة؟
 - هل وسيلة المواصلات لهذه المدينة كان السيارة او الطائرة؟
- يتم مراعاة انشاء مستويين من المطاعم لليوم الواحد.
- في كل مستوى يتم انشاء عقدة لكل مكان ويتم تخزين كافة الخصائص الموجودة في قاعدة البيانات لهذا المكان ضمن هذه العقدة.
- يتم انشاء حافة بين كل عقدة في المستوى وعقد المستوى الذي يليه وهذه الحافة تكون موجهة وذات وزن يمثل المسافة بين العقدتين ويتم حساب هذه المسافة وفق قانون هافرساين وأيضا نقوم بتحديد وسيلة المواصلات المخصصة لهذا المكان بناء على المسافة المحسوبة.
- قانون هافرساين هو معادلة تستخدم في الملاحة وتساعد في حساب البعد الأقصى بين نقطتين على سطح الأرض باستخدام الإحداثيات الجغرافية (خطوط العرض وخطوط الطول) المعبر عنها بوحدة الدرجات. هذه المعادلة مفيدة جدًا في تحديد المسافات الجوية أو البحرية.

$$\text{distance} = 2 * r * \arcsin(\sqrt{(\sin^2(\Delta\varphi / 2) + \cos(\varphi_1) * \cos(\varphi_2) * \sin^2(\Delta\lambda / 2))})$$

حيث:

- "distance" هي المسافة بين النقطتين (تقاس بالكيلومترات).
- "r" هو نصف قطر الأرض وتقديره حوالي 6371 كم.
- φ_1 و φ_2 هما خطوط العرض لل نقطتين الأولى والثانية بالراديان.
- $\Delta\varphi$ هو الفرق في خطوط العرض بين النقطتين بالراديان.
- $\Delta\lambda$ هو الفرق في خطوط الطول بين النقطتين بالراديان.

قانون هافرساين يُعتبر شديد الدقة لحساب المسافات، لاسيما في مساحات الأرض الكبيرة، وذلك بفضل أخذة بعين الاعتبار انحناء سطح الأرض.

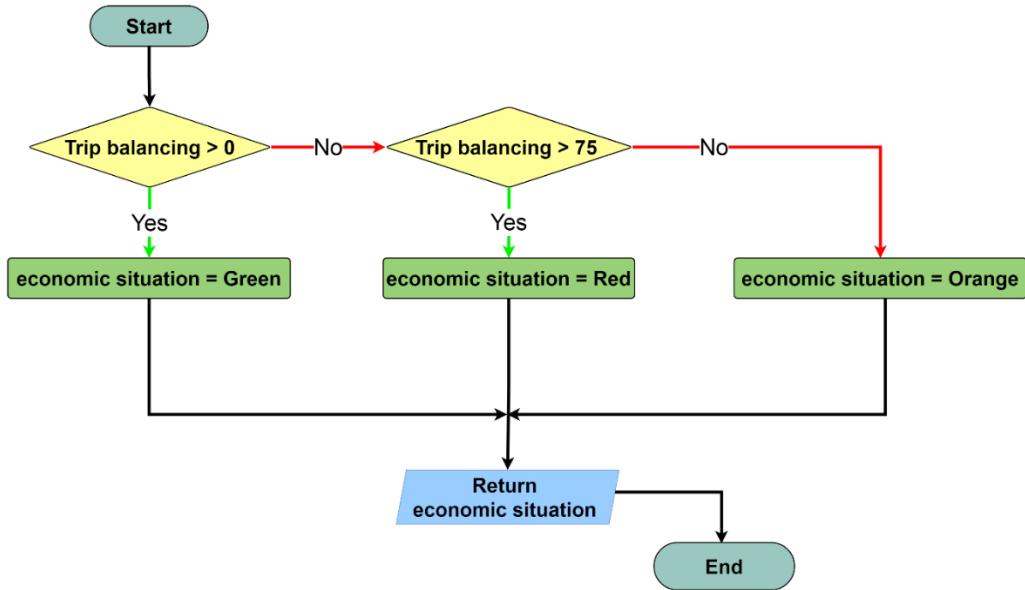
► بعد ذلك نقوم باستدعاء تابع حساب المسار الأقصر المعرف مسبقا في الصيغة $(DijkstraAlghorithm)$ الذي يمثل أيضا أحد الخدمات التابعة للمتحكم $(GenerateTripController)$.

► بعد الحصول على مسار هذا اليوم يتم استخراج خصائص كل مكان وتخزين اسم هذا المكان ضمن مصفوفة الأماكن التي تم زيارتها لكي يتم استبعاده في الأيام التالية المتبقية لهذه المدينة (ضمان عدم تكرر الأماكن).

► يتم حساب كلفة اليوم عن طريق جداء السعر الخاص بكل مكان بعدد الأشخاص مع اضافة كلفة المواصلات بين كل مكان واخر مع مراعاة وسيلة المواصلات التي تم تحديدها سابقا وتخزينها ك احد خصائص الحافة في الصيغة الخاصة ببناء المبيان.

► بعد حساب الكلفة الفعلية للاليوم يتم حساب الفرق بين الكلفة المقدرة لهذا اليوم والكلفة الفعلية ويتم اضافة الناتج الى المتغير $Trip_balancing$, بعد ذلك يتم الحصول على قيمة المؤشر الخاص بالحالة الاقتصادية من أجل تقدير الميزانية المخصصة وعدد الأماكن التي سيتم زيارتها في اليوم التالي.

- قيمة هذا المؤشر يتم ايجادها وفق الشكل (2-3) الذي يمثل المخطط التدفقي لهذه الخطوة.

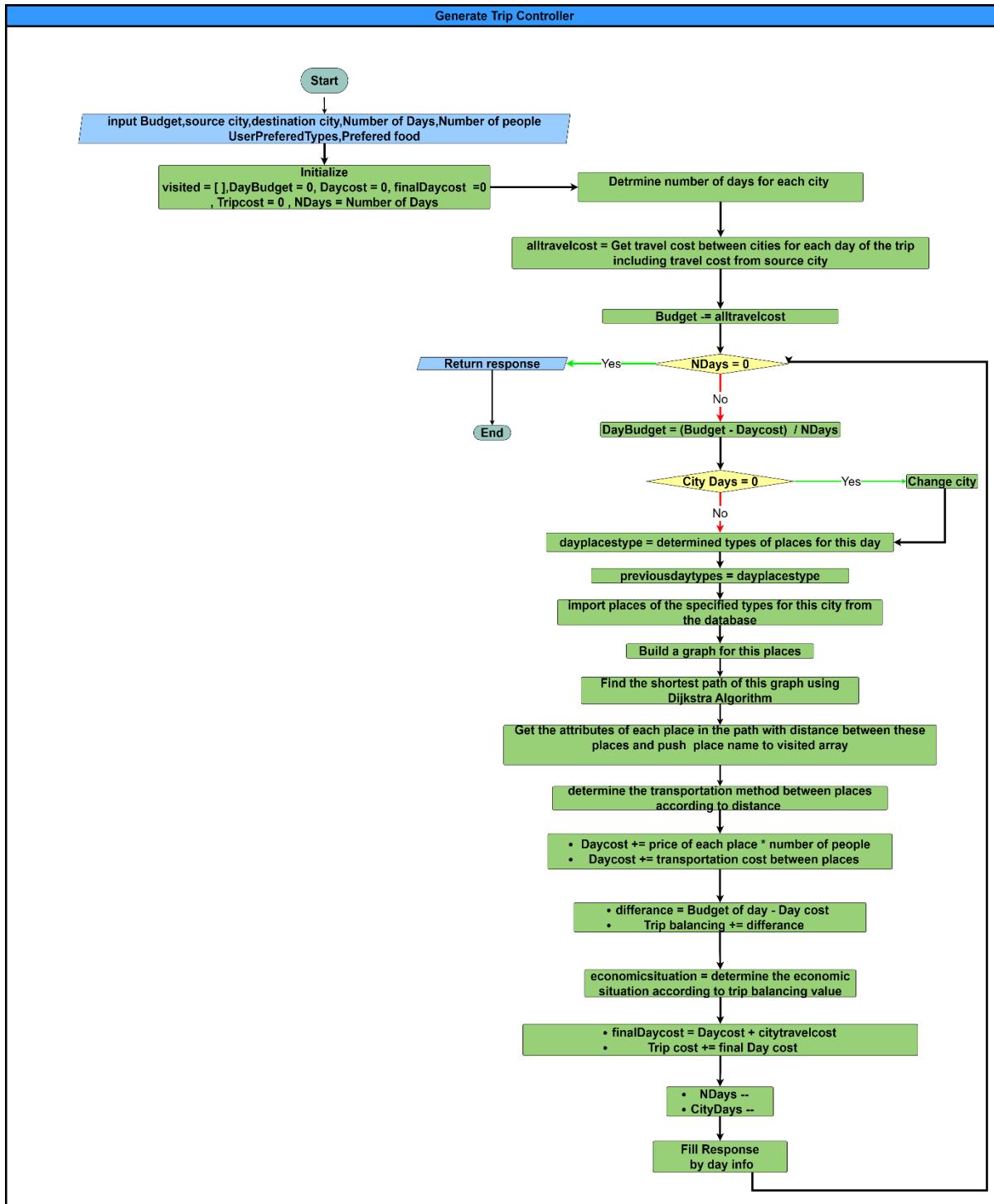


الشكل 2-3: المخطط التدفقى للحالة الاقتصادية

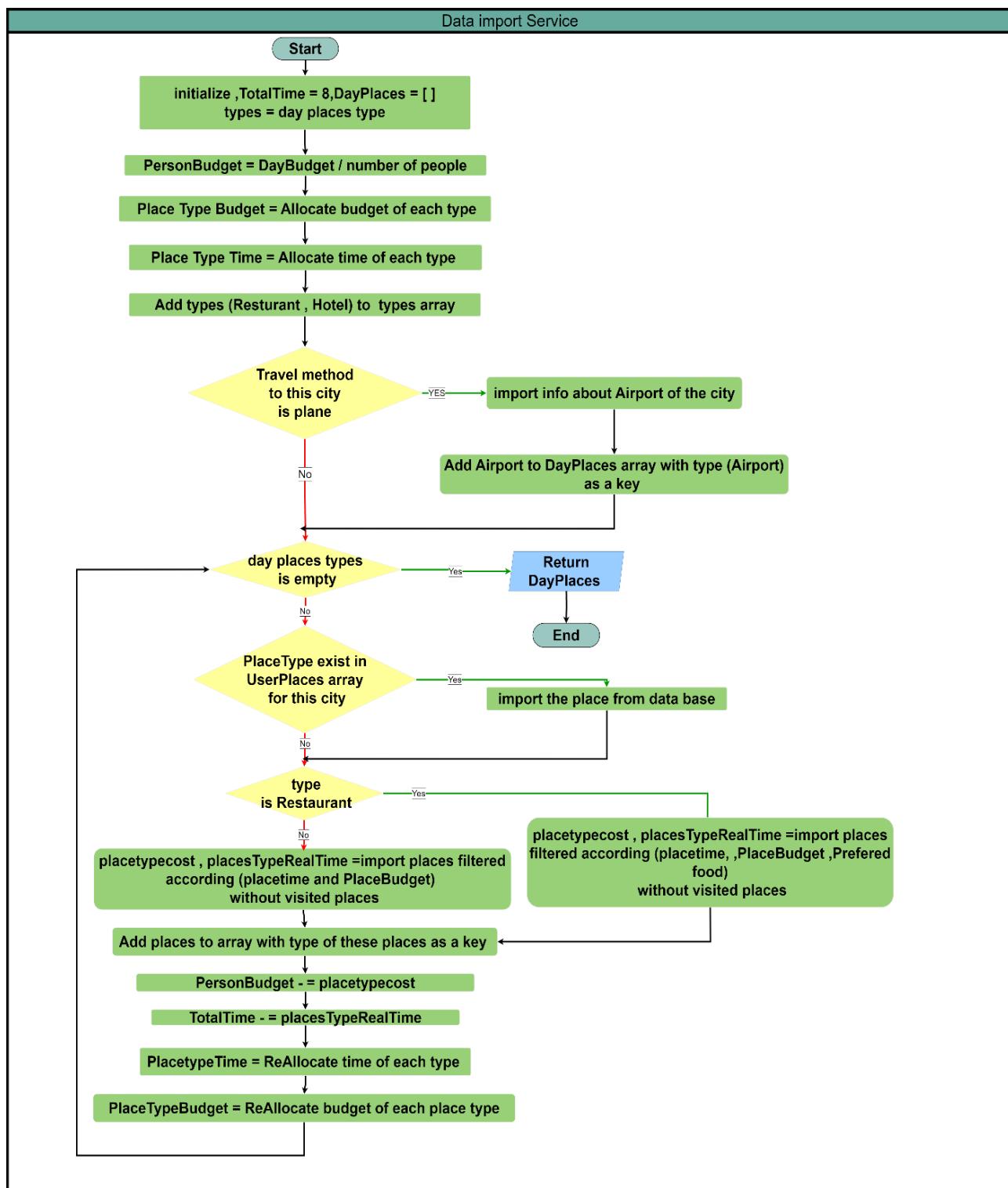
بعد تكرار هذه الخطوات لجميع الأيام يتم حساب الكلفة النهائية للرحلة ويعيد الناتج استجابة تحوي على مسار لكل يوم مع خصائص كل مكان بالإضافة لمسافة المسافة اللازم قطعها للوصول اليه ووسيلة المواصلات الازمة والوقت المخصص لهذا المكان كما تحوي هذه الاستجابة على حجوزات الطيران والحجوزات الفندقية الازمة.

Flowchart diagrams -2-2-2

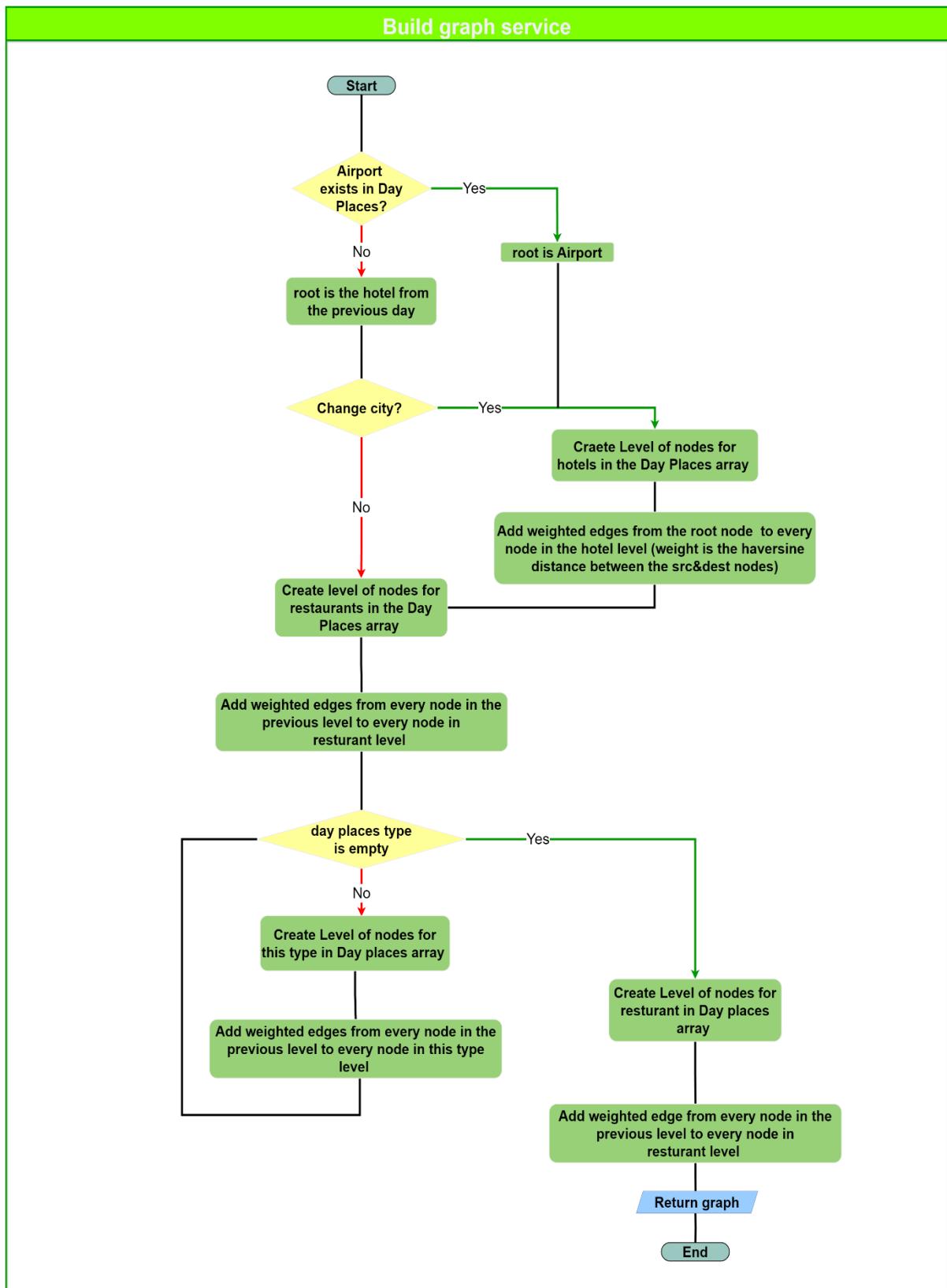
يبين الشكل (4-2) المخطط التدفقي لتوليد الرحلة.



الشكل 4-2: المخطط التدفقي لتوليد الرحلة



الشكل 2-5: المخطط التدفقي لجذب البيانات

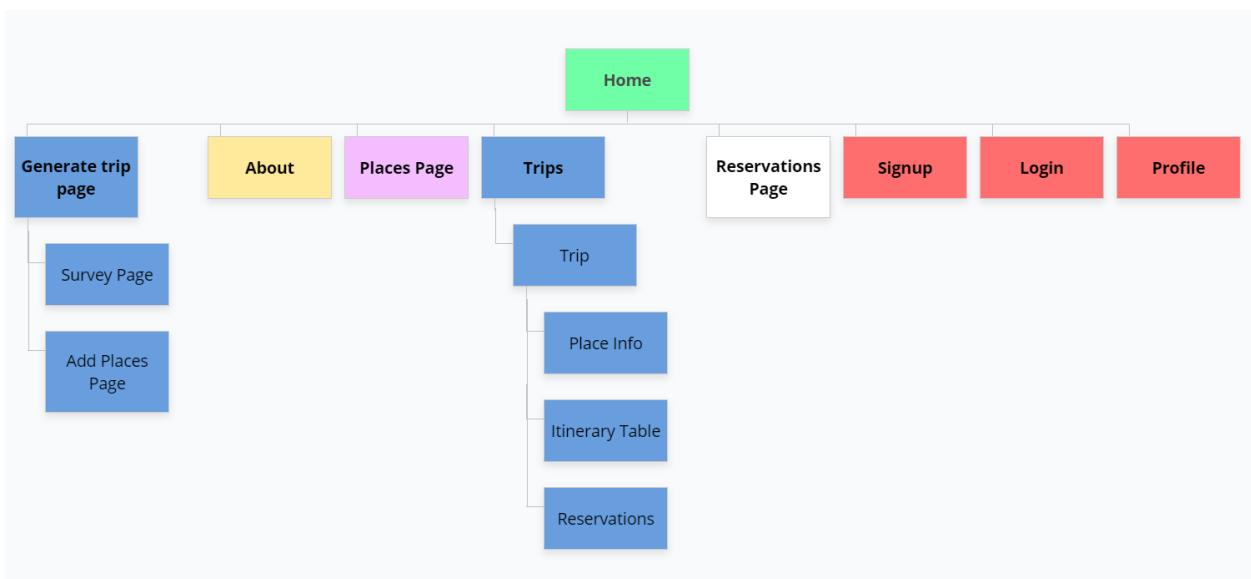


الشكل 2-6: المخطط التدفقى لبناء المبيان

:Site Map Diagram -3-2-2

مخطط خريطة الموقع، المعروف أيضًا بمخطط تسلسل الموقع أو مخطط هيكل الموقع، هو تمثيل بصري لهيكل وتنظيم موقع الويب. يوفر نظرة عامة على مستوى عالٍ لصفحات الموقع وأقسامه وعلاقتها.

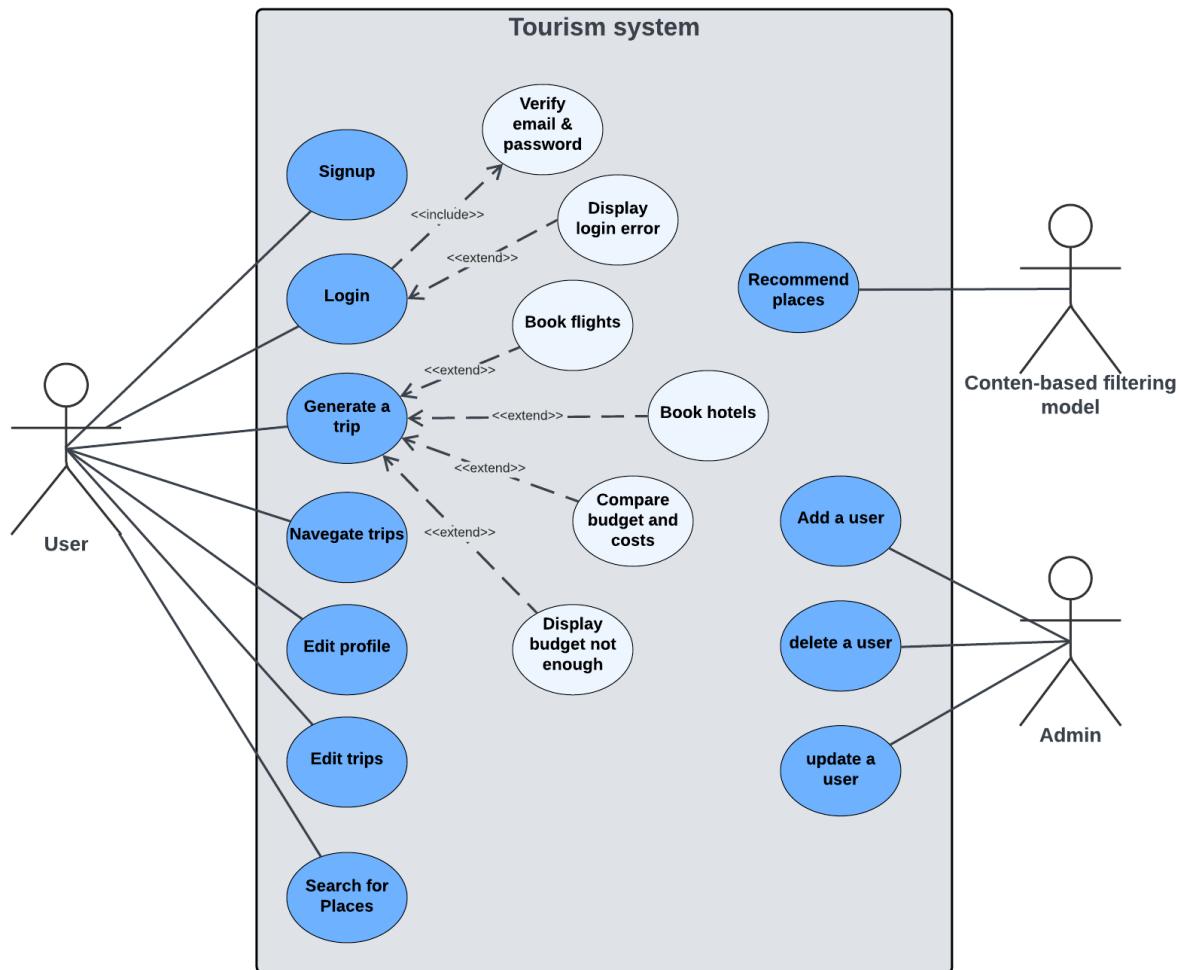
يبين الشكل (7-2) مخطط خريطة الموقع للنظام المطور.



الشكل 7-2 site map diagram :

٤-٢-٢- مخطط حالة الاستخدام (Use Case Diagram)

يبين الشكل (8-2) مخطط حالة الاستخدام للنظام.



الشكل 2-8: مخطط حالة الاستخدام

الفاعلين في النظام:

المستخدم، المدير ونظام الاقتراح.

- حالات الاستخدام الخاصة بالمستخدم:

إنشاء حساب، تسجيل الدخول، توليد رحلة، تصفح الرحل السابقة، التعديل على ملفه الشخصي، التعديل على الرحلة، البحث عن أماكن.

- حالات الاستخدام الخاصة بالمدير:

إضافة مستخدم، حذف مستخدم والتعديل على مستخدم.

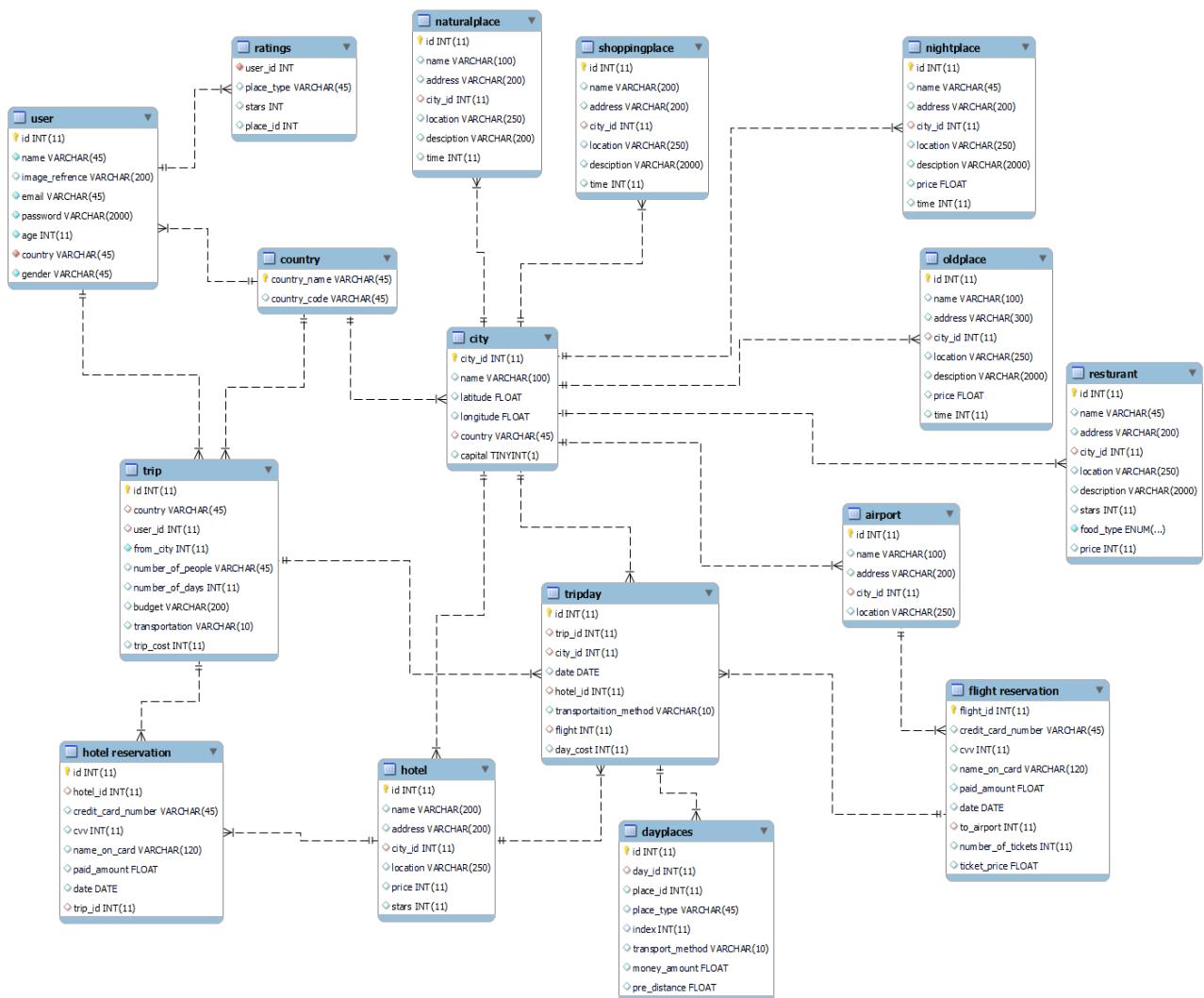
- حالات الاستخدام الخاصة بنظام الاقتراح:

اقتراح أماكن للمستخدم.

EERD Diagram -5-2-2

يستخدم مخطط ال EERD في هندسة البرمجيات وتصميم قواعد البيانات لتمثيل العلاقات بين الكيانات في النظام.

يبين الشكل (9-2) مخطط ال EERD للنظام.



EERD Diagram :9-2

تم في هذا الفصل وضع جميع الخطوات والمخططات الالزame للبدء بمرحلة التنفيذ.

الفصل الثالث: نظام الاقتراح

بني نظام الاقتراح بالاعتماد على خوارزمية content-based filtering, وتم تطبيق هذه الخوارزمية باستخدام الشبكات العصبية والتعلم العميق.

:(dataset) 1-3 البيانات

تقسم البيانات إلى ثلاثة ملفات csv كالتالي:

:users.csv -1

○ يحوي معلومات المستخدمين ويكون من 500 مستخدم.

○ كل مستخدم له الخصائص التالية:

	id	user_name	email	gender	age	country
0	1	Sparhawk	wsparhawk0@va.gov	Female	37	Brazil
1	2	Albery	talbery1@goo.gl	Male	31	Cuba
2	3	Andresen	fandresen2@jiathis.com	Male	44	Bosnia and Herzegovina
3	4	Tansill	stansill3@salon.com	Male	27	South Africa
4	5	Jeffels	ojeffels4@dedecms.com	Female	52	China

:places.csv -2

○ يحوي معلومات عن 685 مكان سياحي.

○ كل مكان سياحي له الخصائص التالية:

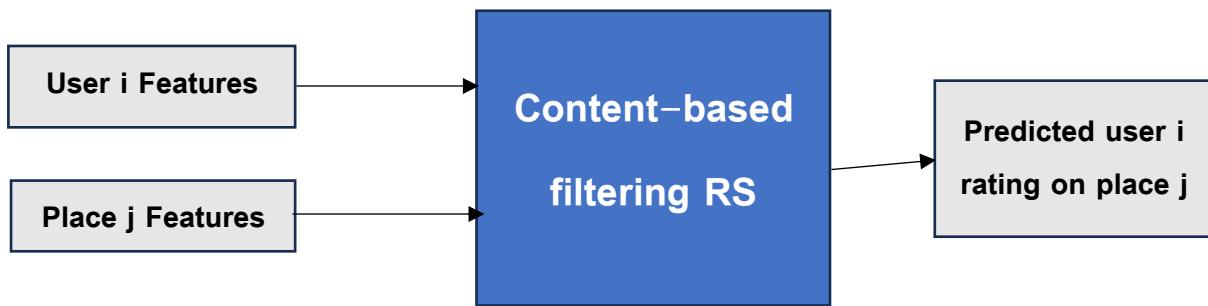
	id	name	address	city_id	location	description	time	stars	food_type	price	place_type	Avg rating
0	1	Galleria Alberto Sordi	Piazza Colonna 00187 Roma RM, Italy	1	41.90122340938924 12.480575693552845	Galleria Alberto Sordi is a historic shopping ...	2.0	NaN	NaN	0.0	shopping	4.101695
1	2	Via del Corso	Via del Corso, 00186 Roma RM, Italy	1	41.903369974944845 12.479473708895839	Via del Corso is one of the main shopping stre...	3.0	NaN	NaN	0.0	shopping	3.784314
2	3	Via Condotti	Via dei Condotti, 00187 Roma RM, Italy	1	41.90533403993258 12.480541881910383	Via Condotti is a prestigious street in Rome, ...	4.0	NaN	NaN	0.0	shopping	4.074074
3	4	La Rinascente	Via del Tritone, 61, 00187 Roma RM, Italy	1	41.90600659996459 12.484760359092748	La Rinascente is a famous department store in ...	4.0	NaN	NaN	0.0	shopping	3.785714
4	5	Porta Portese Market	Via di Porta Portese, 00153 Roma RM, Italy	1	41.87621365001004 12.46784420464272	Porta Portese Market is one of the largest ...	5.0	NaN	NaN	0.0	shopping	4.192982

:ratings.csv -3

○ يحوي تقييمات المستخدمين ويكون من 34544 تقييم.

	User ID	Place ID	Rating
0	1	13	4
1	1	47	5
2	1	58	4
3	1	80	5
4	1	84	4

يبين الشكل (6) المخطط الصندوقي لنظام الاقتراح.



لشكل 3-1: المخطط الصندوقي لنظام الاقتراح.

3-2- تحديد سمات نظام الاقتراح (Recommender system features)

سمات المستخدم (user's features)

- 1- عمر المستخدم.
- 2- جنس المستخدم (ذكر أو انثى).
- 3- بلد المستخدم.
- 4- متوسط تقييم المستخدم للاماكن الاثرية.
- 5- متوسط تقييم المستخدم للاماكن الليلية.
- 6- متوسط تقييم المستخدم للاماكن الطبيعية.
- 7- متوسط تقييم المستخدم لاماكن التسوق.
- 8- متوسط تقييم المستخدم للمطاعم.
- 9- متوسط تقييم المستخدم للفنادق.

سمات المكان (place's features)

- 1- نوع المكان.
- 2- متوسط تقييم المكان.
- 3- تكلفة المكان (سعر المكان).

3-3- المعالجة المسقة للبيانات:

- التعامل مع القيم المفقودة: بعض الأماكن لا تحوي قيمة سعر فتم تعبيتها بأصفار.
- ترميز البيانات (تحويل السمات إلى تمثيلات رقمية): تم تحويل كل من جنس المستخدم وبلده ونوع المكان من نص إلى رقم باستخدام ال `LabelEncoder()` من مكتبة `sklearn`. مثال: `Male ==> 0, Female ==> 1`
- تقييس البيانات (**Scaling**):
 - هي عملية تحويل قيم السمات إلى مقاييس مماثل، مما يضمن مساهمة جميع الميزات بالتساوي في النموذج.
 - تم استخدام ال `StandardScaler()` من مكتبة `sklearn` للقيام بهذه العملية.

3-4- تقسيم البيانات لتدريب واختبار:

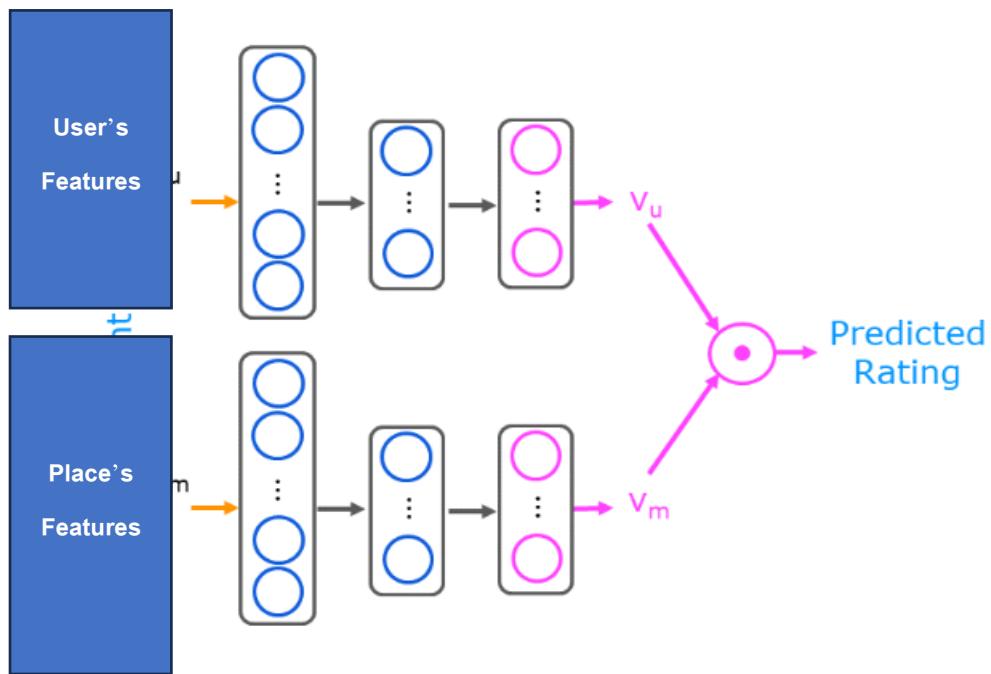
تم تقسيم البيانات المكونة من 34544 record إلى 80% لتدريب النموذج و20% لاختباره.

3-5- بنية الشبكة العصبية:

تم محاولة بناء النموذج بثلاث طرق وتم اختيار الطريقة التي حققت أفضل النتائج.

3-5-1- خوارزمية **content-based filtering** والشبكات العصبية:

- تم تصميم نموذج مكون من شبكتين عصبيتين الأولى تأخذ سمات المستخدم كدخل والثانية تأخذ سمات المكان كدخل، خرج كل من الشبكتين عبارة عن شعاع وناتج ضرب الشعاعين يعطي التقييم المتوقع لهذا المستخدم على هذا المكان.
- طبقات كل شبكة:
 - طبقة `Dense` مكونة من 128 نيرون.
 - طبقة `Dropout` بنسبة 20%.
 - طبقة `Dense` مكونة من 64 نيرون.
 - طبقة `Dropout` بنسبة 20%.
 - طبقة `Dense` مكونة من 64 نيرون.



الشكل 3-0: معمارية الشبكة العصبية المستخدمة في خوارزمية *content-based filtering*

25.66% :Loss value

:Tensorflow code

```

user_NN = tf.keras.models.Sequential([
    keras.layers.Dense(128 ,activation='relu'),
    keras.layers.Dropout(0.2),
    keras.layers.Dense(64 ,activation='relu'),
    keras.layers.Dropout(0.2),
    keras.layers.Dense(32)
])
item_NN = tf.keras.models.Sequential([
    keras.layers.Dense(128 ,activation='relu'),
    keras.layers.Dropout(0.2),
    keras.layers.Dense(64 ,activation='relu'),
    keras.layers.Dropout(0.2),
    keras.layers.Dense(32)
])
# create the user input and point to the base network
input_user = tf.keras.layers.Input(shape=(9))
vu = user_NN(input_user)
vu = tf.linalg.l2_normalize(vu, axis=1)
# create the item input and point to the base network
input_item = tf.keras.layers.Input(shape=(3))
vm = item_NN(input_item)
vm = tf.linalg.l2_normalize(vm, axis=1)
# compute the dot product of the two vectors vu and vm
output = tf.keras.layers.Dot(axes=1)([vu, vm])

```

2-5-3- التوقع (regression) والشبكات العصبية (Dense)

طبقات الشبكة:

1. طبقة Dense مكونة من 256 نيرون.
2. طبقة Dropout بنسبة 20%.
3. طبقة Dense مكونة من 128 نيرون.
4. طبقة Dropout بنسبة 20%.
5. طبقة Dense مكونة من 64 نيرون.
6. طبقة Dense مكونة من نيرون واحد.

28.31% :Loss value

:Tensorflow code

```
model = Sequential()  
model.add(Dense(256, activation='relu', input_dim=X_scaled.shape[1]))  
model.add(Dropout(0.2))  
model.add(Dense(128, activation='relu'))  
model.add(Dropout(0.2))  
model.add(Dense(64, activation='relu'))  
model.add(Dense(1))
```

3-5-3- التوقع (regression) والشبكات العصبية (LSTM)

طبقات الشبكة:

1. طبقة LSTM مكونة من 256 نيرون.
2. طبقة Dropout بنسبة 20%.
3. طبقة LSTM مكونة من 128 نيرون.
4. طبقة Dropout بنسبة 20%.
5. طبقة Dense مكونة من 64 نيرون.
6. طبقة Dense مكونة من نيرون واحد.

28.01% :Loss value

:Tensorflow code ○

```
model = Sequential()
model.add(LSTM(128, input_shape=(time_steps, num_features),
return_sequences=True))
model.add(Dropout(0.2))
model.add(LSTM(64))
model.add(Dropout(0.2))
model.add(Dense(32, activation='relu'))
model.add(Dense(1))
```

4-5-3- المقارنة بين الطرق الثلاث:

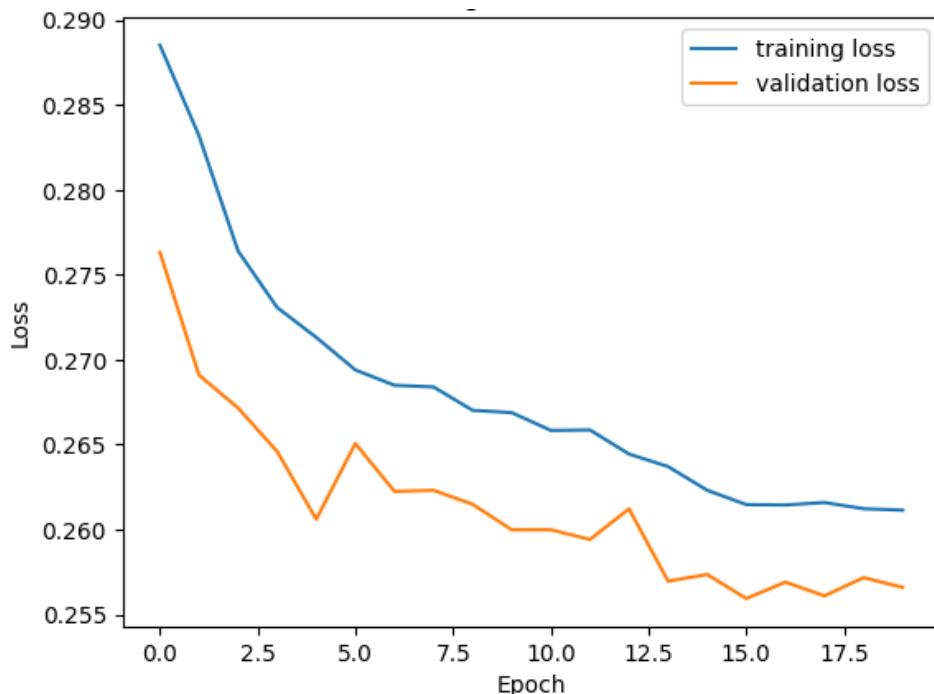
بعد تدريب النماذج الثلاث على نفس البيانات واختبارهم باستخدام بيانات الاختبار وجدنا:

الجدول 3-1: المقارنة بين النماذج

	Dense Layers	LSTM Layers	Content-based Filtering
loss	28.31%	28.01%	25.66%

بالمقارنة بين الطرق الثلاث نجد ان الطريقة الأولى (content-based filtering) هي التي حققت اقل قيمة خطأ (loss) لذلك تم استخدامها.

يبين الشكل (8) تناقص الخطأ أثناء عملية التدريب باستخدام خوارزمية content-based filtering



الشكل 3-3: تغير قيمة الخطأ أثناء عملية التدريب

الفصل الرابع: التطبيق العملي والنتائج

يتضمن هذه الفصل تطبيق الخطوات التي تم تصميمها بالإضافة إلى الأدوات والتقنيات المستخدمة لتطوير الموقع.

1- جمع البيانات:

من المفترض أن يكون النظام قادر على إنشاء خطة سياحية لا يجد في العالم لكن لتحقيق ذلك لا بد من وجود كمية بيانات ضخمة جداً عن الأماكن السياحية والفنادق والمطاعم في جميع أنحاء العالم ونحن لا نملك وصولاً مثل هذه الكمية من البيانات.

النظام في الوقت الحالي قادر فقط على إنشاء خطة سياحية لإيطاليا.

تم جمع البيانات حول الأماكن السياحية الموجودة في إيطاليا وهذه الأماكن تتضمن "5" مدن في إيطاليا. المدن هي: روما، ميلان، فينيس، فلورانس، ونابولي.

أنواع الأماكن هي (فنادق، مطاعم، أماكن تسوق، أماكن ليلية، أماكن أثرية، وأماكن طبيعية) وبالمجمل تم جمع معلومات عن 685 مكان.

البيانات المجمعة عن كل مكان:

1- اسم المكان

2- عنوان المكان

3- احداثيات المكان (longitude and latitude)

4- وصف المكان: سطر أو بضعة أسطر تتحدث عن المكان بشكل عام

5- نوع الطعام: إذا كان المكان مطعماً فسيكون له خاصية نوع الطعام (تقليدي، بحري، طعام سريع، طعام فاخر، وحلويات)

6- متوسط تقييم المكان (عدد النجوم من 1 إلى 5)

بالإضافة إلى هذه البيانات المجمعة تم توليد بعض البيانات بشكل عشوائي لعدم قدرتنا على جمعها، مثل:

1- متوسط عدد الساعات الالزمة لمشاهدة الأماكن الأثرية فليس كل الأماكن الأثرية يمكن مشاهتها بشكل كامل في ساعة واحدة وليس لدينا إمكانية الوصول لمثل هذه البيانات.

2- تكلفة المكان: إذا كان مطعماً فتكلفته المكان هي متوسط سعر الوجبة لشخص واحد، وإذا كان فندقاً فتكلفته المكان هي تكلفة الغرفة لشخص واحد، وإذا كان مكاناً أثرياً أو ليليًّا ف فهي سعر التذكرة، أما الأماكن الطبيعية فهي مجانية.

مثال عن بيانات احد المطاعم:

name	address	description	latitude	longitude	Food type	stars	price
Roscioli	Via dei Giubbonari ,21,00186 Roma RM, Italy	Roscioli is an award-winning fine dining restaurant situated on Via dei Giubbonari...	41.894	12.473	Fine-dinning	5	50

مثال عن بيانات احد الفنادق:

name	address	latitude	longitude	stars	price
Hotel Quirinale	Via Nazionale 7, Central Station, 00184 Rome, Italy	41.901	12.494	5	250

4-2- الواجهات الخلفية (Back-end)

1-2-4 - اللغات المستخدمة:

1- لغة PHP:

• هي لغة برمجة شائعة جداً ومحصصة بشكل أساسي لتطوير الويب. تتميز بنمطها التقسيري وكونها سهلة الاستخدام مع دعم واسع لقواعد البيانات وملفات النظام، مما يجعلها مناسبة جداً لإنتاج موقع ويب ديناميكية. من ناحية التطبيق، PHP متوافقة مع العديد من خوادم الويب وأنظمة التشغيل. ما يميز PHP أيضاً هو رفيقها القوي، سيرفر MySQL، مما يوفر لمطوري الويب بيئة متكاملة لتطوير تطبيقات ويب مرنّة وموثوقة. على الرغم من كونها لغة برمجة ناضجة، إلا أن PHP ما زالت تتطور باستمرار، مع إضافة ميزات جديدة وتحسينات في كل إصدار، مما يحافظ على مكانتها كأداة قيمة في تطوير الويب.

• مميزات لغة PHP:

1- سهولة الاستخدام.

2- مجانية ومفتوحة المصدر.

3- القدرة على التفاعل مع قواعد البيانات.

- 4- دعم ضخم من المجتمع.
- 5- مدمجة بخوادم الويب.
- 6- أداء وكفاءة ممتازين.
- استخدام لغة PHP: لبدء استخدام PHP، تحتاج أولاً إلى إعداد بيئة تطوير تشمل تثبيت خادم ويب مثل Apache أو Nginx، وتنصيب PHP نفسه. ويمكن أيضاً استخدام حزم مثل XAMPP أو MAMP التي تسهل عملية التثبيت بتوفير بيئة متكاملة.

- 2-2-4 - **إطار عمل Laravel**

إطار عمل Laravel هو منصة مفتوحة المصدر مصممة لتطوير تطبيقات الويب بلغة PHP. يتميز ب أناقة الكود والبساطة، ويوفر الكثير من المزايا مثل نظام توجيه طلبات متقدم وقوى، دعم للاختبار الآلي. Laravel يتبع هندسة MVC، مما يجعل الكود منظماً وقابلً للصيانة بشكل أكبر. واجهة خلفية Laravel يقدم مجموعة من الأدوات التي تسهل عملية التطوير مثل Eloquent ORM للتعامل مع قواعد البيانات، مجموعة واسعة من المكتبات الداخلية، ومدير حزم Composer لإدارة الاعتمادات. باستخدام Laravel في الباك اند فقط، يتقلص دور View، ويكون التركيز أكثر على بناء API قوية وأمنة. يسهل هذا من خلال دعمه لمعايير API مثل REST و GraphQL، ويوفر جاهزية عالية لشفير البيانات وحمايتها عبر ميكانيكيات الأمان المتقدمة كخوارزميات تشفير معتمدة ومحدثة. الأداء مع Laravel كتطوير واجهة خلفية يُعتبر مرضي، ويمكن تحسينه أكثر باستخدام أدوات مثل كاش التطبيق وتقنية queues لتسريع معالجة العمليات. Laravel يُعزز من كفاءة التطوير بفضل نظام الترحيل المبني داخلياً لتعديل وإدارة قواعد البيانات بسلاسة.

:Laravel Sanctum - 1

- Laravel Sanctum هو أداة مساعدة تُضاف إلى تطبيقات Laravel لمساعدتها على إدارة عمليات تسجيل الدخول وتوثيق هوية المستخدمين من خلال شهادات (tokens).
- تم تصميمه خصيصاً للتطبيقات التي تعمل على صفحة واحدة (- Single Page Applications) والتطبيقات المحمولة وواجهات برمجة التطبيقات (APIs) التي لا تحتاج إلى تعقيدات نظام SPA لـ OAuth للتوثيق.

- Sanctum يُستطيع أن يعطي كل مستخدم خاصية دخول فريدة (token)، وهذه الشهادة يمكن استخدامها للتحقق من هوية المستخدم في كل مرة يحتاج فيها إلى الوصول إلى الخدمة. بمعنى آخر، يمنحك "مفتاح" للدخول لتطبيقك.
- من أهم ميزات Sanctum : من أهم ميزات Sanctum
- 1- دعم SPA: يوفر طريقة سهلة وآمنة لإدارة الجلسات وتوثيق المستخدمين في التطبيقات التي تعمل على صفحة واحدة.
- 2- حماية ضد هجمات CSRF: يساعد على حماية الطلبات من نوعية هجمات الـ Request Forgery.
- 3- سهل الدمج: يمكن إضافته إلى أي مشروع Laravel ببساطة دون الحاجة إلى إعادة كتابة أجزاء كبيرة من التطبيق.
- 4- توثيق فردي لكل مستخدم: بإمكان كل مستخدم أن يحصل على شهادة دخول خاصة به تسمح بتتبع أكثر أماناً ودقة.

- باختصار، Laravel Sanctum يسهل عملية توثيق المستخدمين والتحقق من هويتهم، خاصة في التطبيقات الحديثة وواجهات برمجة التطبيقات، بطريقة بسيطة وآمنة ولاستخدامها ضمن التطبيق نقوم بكتابة في ملف routes/api.php مايلي:

```
Route::middleware('auth:sanctum')->get('/user', function
(Request $request) {
    return $request->user();
});
```

```
Route::put('/create_trip', TripController::class .
'@create_trip')
->middleware('auth:sanctum');
```

- 2- استكمال بيئة التطوير:
 - لاستكمال بيئة التطوير لدينا بهدف استخدام إطار العمل Laravel نحتاج أولاً لتنصيب نظام إدارة الحزم composer فهو أداة تشكل الأساس لمعظم تطوير PHP الحديث. Composer هو مدير

التعبيات لـ PHP، شبيه بـ Node لـ NPM أو RubyGems لـ Ruby. لكن مثل Composer أيضًا يشكل أساس الكثير من اختبرتنا، تحميل السكريبتات المحلية، سكريبتات التنصيب، والكثير غير ذلك. ستحتاج إلى Composer لتنصيب Laravel، تحديث Composer، وإضافة التعبيات الخارجية [10].

- نقوم بإنشاء مشروع Laravel جديد باستخدام CLI (واجهة سطر الأوامر):

```
composer create-project laravel/laravel example-app
```

- نقوم بإنشاء نموذج لكل جدول في قاعدة البيانات. النموذج يمثل البيانات ويمكن استخدامه للتعامل معها:

```
php artisan make::model modelname
```

- Artisan: مجموعة من الأوامر المدمجة للتشغيل عبر سطر الأوامر مع القدرة على إضافة المزيد.
- إنشاء توجيهات لـ API: في ملف routes/api.php، قم بتحديد توجيهات لـ API، مثلًا:

```
Route::post('/signup', UserController::class .  
'@signup');
```

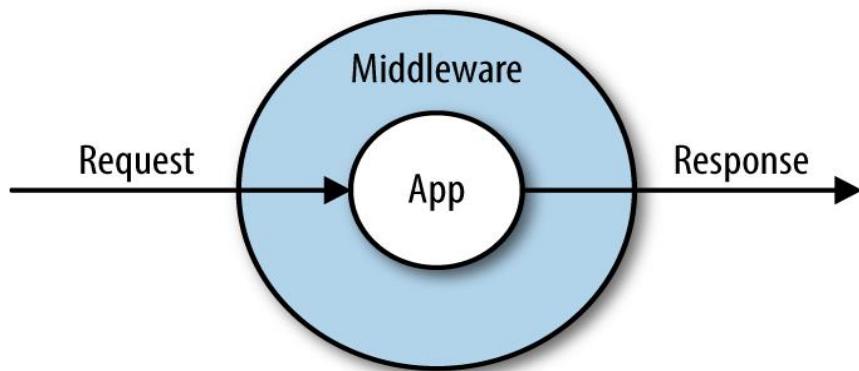
- من خلال هذا الملف يتم توجيه رسالة الطلب إلى المتحكم المناسب ويتتيح بناء نقاط النهاية لـ API تتوافق مع الأساليب القياسية لـ HTTP مثل GET، POST، PUT، DELETE (endpoints) وغيرها.
- إنشاء المتحكمات (Controllers):

```
php artisan make::controller controllername
```

- الـ Controllers في إطار عمل Laravel تعمل ك وسيط بين المستخدم والتطبيق، حيث تولى مهمة استقبال طلبات المستخدم، معالجة هذه الطلبات بالتعاون مع الموديلات (نماذج البيانات) والخدمات

الأخرى المستخدمة داخل التطبيق، ثم تحديد الرد المناسب الذي يجب عودته إلى المستخدم، سواء كان ذلك على شكل عرض صفحة ويب، JSON، أو أي نوع آخر من الاستجابات. بهذه الطريقة، تساهم في تنظيم منطق العمل وفصل الأدوار داخل التطبيق.

- **Middleware**: تستخدم للتعامل مع القضايا العرضية مثل **authentication** وهو طبقة تقع بين الطلب (**request**) والرد (**response**)، وتسخدم لفلترة الطلبات وتنفيذ بعض العمليات قبل وصولها إلى الراوتر (**Router**) أو المتحكم (**Controller**). يمكن استخدامها للتحقق من الصلاحيات، توجيه المستخدمين بناءً على حالة تسجيل الدخول، تعديل الطلبات أو الردود، وغيرها من الوظائف التي تحتاج إلى التنفيذ قبل أو بعد معالجة الطلب الفعلية. هذه الطبقة توفر آلية مرنة لإضافة الأمان والتحكم في تدفق البيانات داخل التطبيق بطريقة منظمة.



الشكل 4-1: دورة حياة الطلب-الاستجابة

3- التعامل مع قاعدة البيانات:

- يتم التعامل مع قاعدة البيانات وفق ما يلي [10][11]:
 - 1- التهيئة: تبدأ بتهيئة إعدادات قاعدة البيانات في ملف `.env`. يتيح Laravel تحديد إعدادات الاتصال لمختلف أنواع قواعد البيانات تحت قسم `database` في ملف `database.php`.
 - 2- الهجرات: يُستخدم نظام الهجرات في Laravel لتعريف قاعدة البيانات وجداولها وعلاقتها بواسطة PHP بدلًا من كتابة خطوط SQL يدوياً. الهجرات تُبسط عملية إنشاء وتعديل جداول قاعدة البيانات وتحل إمكانية تتبع الإصدارات والتعاون بين فريق العمل.

PHP (Models) : Eloquent ORM -3

تمثل جداول قواعد البيانات. يتيح لك ORM إجراء عمليات CRUD والتعامل مع العلاقات بين الجداول

بسهولة.

Query Builder -4 : للتعامل مع الاستعلامات المعقّدة، يقدم Laravel واجهة Query Builder التي

تسمح بتكوين استعلامات SQL بطريقة سلسة وقابلة القراءة في PHP.

- توفر هذه الأدوات معاً إطار عمل قوي يجعل العملية الإجمالية للتعامل مع قاعدة البيانات فعالة و مباشرة، بالإضافة إلى تحسين جودة كود التطبيق وسهولة صيانته.
- تظهر النتائج عموماً أن Query Builder يكون أسرع في أداء الاستعلامات مقارنة ب Eloquent ORM. السرعة في هذا السياق تُعزى إلى أن Query Builder يعتبر واجهة أقل تجريدًا لقاعدة البيانات؛ حيث يتعامل مباشرة مع الاستعلامات دون الحاجة إلى الكثير من المعالجات الإضافية التي تتضمنها Eloquent، مثل تحليل العلاقات بين النماذج أو تحميل البيانات تلقائياً لكن يجب الأخذ في الاعتبار أن الاختيار بين استخدام Eloquent أو Query Builder لا يجب أن يعتمد على السرعة فقط. Eloquent يقدم مزايا برمجية عديدة من حيث بساطة الرمز، إمكانية قراءته وصيانته، ونظام قوي لإدارة العلاقات بين الجداول في قاعدة البيانات [10].
- تم استخدام كلا الطريقيتين Eloquent ORM وQuery Builder في التعامل مع قاعدة البيانات في تطوير هذا الموقف وذلك لتحقيق التوافق بين ففي الاستعلامات المعقّدة تم اللجوء إلى Query Builder أما في الاستعلامات البسيطة تم استخدام Eloquent ORM وذلك من أجل تحقيق التوافق بين سرعة الأداء وبين بساطة الرمز وامكانية صيانته.

4 - مكتبة PHP في Clue Graph :

- هي واحدة من المكتبات التي تقدم حلًّا للتعامل مع البيانات المُنظمَة على شكل رسم بياني (Graph). تُستخدم هذه المكتبة في عمليات مثل إنشاء رسوم بيانية، التنقل بين العقد، وإجراء عمليات حسابية مختلفة على الرسوم البيانية مثل البحث عن أقصر طريق بين نقطتين.
- المميزات الرئيسية لمكتبة Clue Graph :

5- تصميم موجه وغير موجه للرسم البياني: تتيح التعامل مع الرسوم البيانية الموجهة وغير الموجهة، مما يزيد من مرونتها في مختلف سيناريوهات الاستخدام.

6- استكشاف وتقييم البيانات: توفر وظائف للبحث عن عقد أو حواضن محددة، إلى جانب دعم عمليات الإضافة والحذف.

7- مرونة في التمثيل: تتيح مرونة في تمثيل البيانات، مما يجعلها مفيدة في تطبيقات مثل تحليل الشبكات الاجتماعية، إدارة المشاريع، وغيرها من المجالات التي تعتمد على الرسوم البيانية.

- للبدء باستخدام Clue Graph، يتطلب ذلك تثبيت المكتبة عبر Composer، باستخدام الأمر التالي:

```
composer require clue/graph
```

- بعد التثبيت، يمكن إنشاء كائن رسم بياني والبدء بإضافة العقد والحواف.

5- مكتبة Graph/Graphviz

• هي أداة قوية وشاملة لتصور البيانات المنظمة على شكل رسم بياني. تُستخدم هذه المكتبة لإنشاء تمثيلات رسومية للبيانات المعقدة، بطريقة تجعل تحليل العلاقات بين العقد في الرسم البياني أكثر وضوحاً وسهولة. تُعرف GraphViz ببرونتها وقدرتها على توليد رسوم بيانية بتسميات مختلفة، بما في ذلك PDF، PNG، SVG.

• تم استخدام هذه المكتبة لاختبار صحة توليد الرسم البياني وبناء المستويات بالشكل المطلوب وللبدء باستخدامها يتطلب ذلك تثبيت المكتبة عبر Composer وبعد ذلك تثبيت برنامج GraphViz ومن ثم استخدام التعليمات التالية من أجل ظهار الرسم:

```
$graphviz = new GraphViz(['binary' => 'C:\Program  
Files\Graphviz\bin\dot.exe']); // for display the created graph  
$graphviz->display($graph2);
```

4-3-2- البرامج المستخدمة:

1- برنامج XAMPP

• هو حزمة برمجيات حرة تهدف لتسهيل نشر خادم ويب محلي يشمل PHP، MySQL، Apache، MySQL.

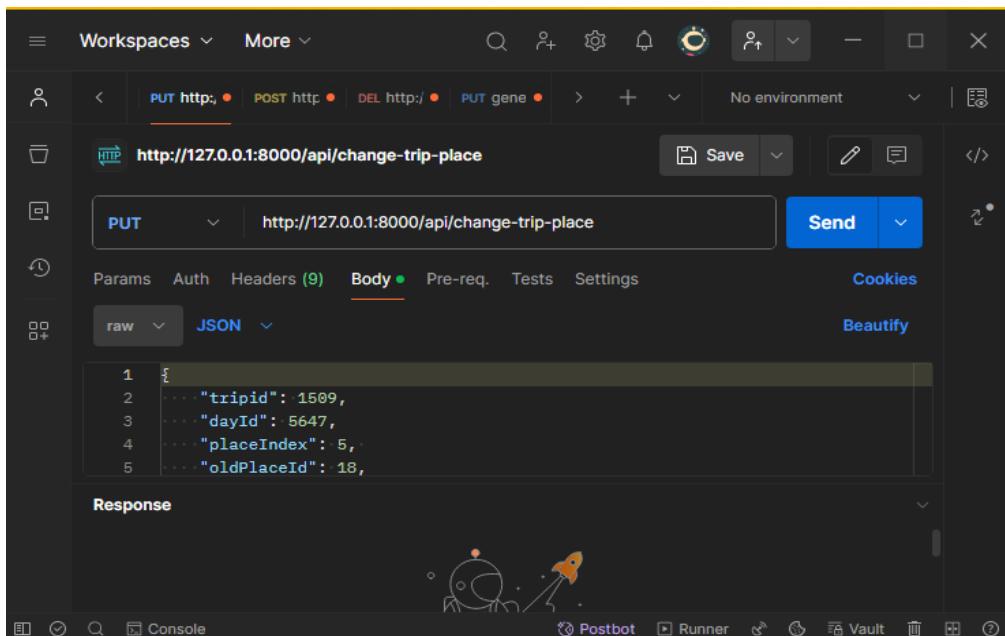
• يستخدم بشكل كبير بين مطوري الويب لإنشاء واختبار موقع وتطبيقات الويب بيئياً على الأجهزة الشخصية قبل النشر على الإنترنت. XAMPP يعطي المطوريين القدرة على تطوير وتجربة السكريبتات والتطبيقات بسهولة من دون الحاجة إلى الوصول إلى خادم ويب عن بعد.



الشكل 4-2: برنامج XAMPP

2- برنامج postman

- هو أداة قوية وشائعة الاستخدام بين مطوري البرمجيات لتصميم، اختبار، وتوثيق واجهات برمجة التطبيقات (APIs). يتيح Postman للمستخدمين إرسال الطلبات إلى الواجهات البرمجية وعرض الاستجابات بطرق سهلة وفعالة، مما يسرع عملية التطوير ويساعد في التأكد من صحة الواجهات البرمجية. يدعم التطبيق استخدام العديد من طرق الطلبات مثل GET، POST، PUT، DELETE، ويوفر واجهة مستخدم مرنة تسمح بتنظيم الطلبات ضمن مجموعات ومشاركتها مع فريق العمل.



الشكل 4-3: برنامج postman

4-3- اللغات والمكتبات المستخدمة في بناء النماذج (models)

Python -1

- Python هي لغة برمجة تفسيرية (interpreted) وغرضية التوجه وتعتبر لغة برمجة عالية المستوى أيضا.
- تعتبر Python لغة متعددة الاستخدامات، حيث يمكن استخدامها في تطوير تطبيقات سطح المكتب وتطبيقات الويب وتطبيقات الهندسة وعلوم البيانات والذكاء الاصطناعي والألعاب والتطبيقات المحمولة، وغيرها الكثير.
- بفضل بساطتها وقوتها، أصبحت Python واحدة من اللغات البرمجية الأكثر شعبية في العالم، وستخدمها العديد من الشركات الكبيرة والمؤسسات الحكومية والمطوريين المستقلين في جميع أنحاء العالم.

TensorFlow -2

مكتبة قوية للحساب العددي ، وهي مناسبة بشكل خاص للتعلم الآلي على نطاق واسع (ولكن يمكن استخدامها لأي شيء آخر يتطلب حسابات تقيلة). تم تطويرها بواسطة فريق Google Brain وهو يعمل على تشغيل العديد من خدمات Google واسعة النطاق. كانت مفتوحة المصدر في نوفمبر 2015، وهي الآن مكتبة التعلم العميق الأكثر شعبية (من حيث الاستشهادات في الأوراق، والبني في الشركات، وعدد النجوم على GitHub، وما إلى ذلك)، مشاريع لا حصر لها تستخدم TensorFlow لجميع أنواع مهام التعلم الآلي، مثل تصنيف الصور ومعالجة اللغات الطبيعية (NLP) وأنظمة التوصية والتنبؤ بالسلسل الزمنية وغيرها.

sklearn -3

مكتبة برمجيات مجانية لتعلم الآلة للغة برمجة بايثون. تتميز بالعديد من خوارزميات التصنيف والانحدار والتجميع وهي مصممة للتفاعل مع مكتبات Python العددية والعلمية NumPy وSciPy. وهي مكتبة سهلة الاستخدام للغاية، ومع ذلك فهي تنفذ العديد من خوارزميات التعلم الآلي بكفاءة.

pandas -4

مكتبة مفتوحة المصدر لتحليل البيانات ومعالجتها مبنية على لغة البرمجة Python. توفر هيكل البيانات والعمليات لمعالجة الحدائق الرقمية والسلسل الزمنية. تستخدم على نطاق واسع لتنظيف البيانات وتحويلها وتحليلها.

matplotlib -5 : مكتبة رسم و تخطيط للغة برمجة Python و ملحقها الرياضي العددي NumPy. توفر واجهة برمجة تطبيقات غرضية التوجه. يتم استخدامها بشكل شائع لإنشاء تصورات ومنحنيات ثابتة أو تفاعلية في Python.

4-4- الواجهة الامامية :Front-end

1-4-4- اللغات المستخدم:

:HTML -1

- HTML (Hyper Text Markup Language) هي الكود المستخدم لهيكلة صفحة ويب و محتواها. على سبيل المثال، يمكن تنظيم المحتوى ضمن مجموعة من الفقرات أو قائمة من النقاط ذات التعداد النقطي أو باستخدام الصور و جداول البيانات.
- HTML هي لغة تتميز تحدد بنية المحتوى الخاص بك. تتكون HTML من سلسلة من العناصر، والتي تستخدمها للإحاطة بأجزاء مختلفة من المحتوى لإظهاره بطريقة معينة، أو لجعله يتصرف بطريقة معينة. يمكن للعلامات المرفقة (enclosing tags) إنشاء ارتباط تشعبي لكلمة أو صورة إلى مكان آخر، ويمكنها كتابة الكلمات بخط مائل، ويمكن أن تجعل الخط أكبر أو أصغر، وما إلى ذلك.

:CSS -2

- CSS (Cascading Style Sheets) هو الرمز الذي يصمم محتوى الويب.
- CSS هي لغة تحدد نمط وثيقة الويب. تحكم في كل شيء من الألوان والخطوط إلى التباعد والتخطيط على صفحة الويب، مما يجعلها تبدو جذابة بصرياً. في الجوهر، تصف CSS كيفية عرض عناصر HTML على الشاشة أو الورق أو وسائل أخرى. على سبيل المثال، يمكنك استخدام CSS للتغيير لون وحجم العناوين، وإنشاء تخطيطات، وحتى إضافة الرسوم المتحركة.

:TypeScript -3

- TypeScript هي مجموعة فرعية من JavaScript تضيف تعليقات النوع (type annotations) إلى JavaScript ، مما يساعد على تجنب الأخطاء المحتملة التي قد تحدث أثناء التشغيل.
- توفر TypeScript أدوات أفضل للتطبيقات الكبيرة ويتم تحويلها إلى JavaScript ، مما يجعلها متوافقة مع المتصفحات.

2-4-4 - المكتبات المستخدمة:

React.js -1

Material UI -2: هي مكتبة مجانية ومفتوحة المصدر لتصميم واجهات المستخدم باستخدام React

تتبع إرشادات تصميم من جوجل وتتوفر مجموعة كبيرة من العناصر الجاهزة للاستخدام مثل الأزرار والقوائم والحقول تستخدم في إنشاء تطبيقات الويب والهاتف المحمول ذات تصميم جذاب وكفاءة عالية.

TomTom Maps -3: مكتبة خرائط TomTom هي مجموعة من أدوات تطوير البرمجيات (SDK) التي توفرها TomTom ، وهي شركة رائدة في مجال تكنولوجيا الموقع وخدمات الخرائط. تتيح المكتبة للمطوريين دمج قدرات الخرائط والتنقل الخاصة بـ TomTom في تطبيقاتهم وخدماتهم الخاصة.

Axios -4: مكتبة JavaScript شهيرة تستخدم لإجراء طلبات HTTP من متصفحات الويب، توفر واجهة برمجة التطبيقات (API) بسيطة وبديهية تسمح للمطوريين بإرسال طلبات HTTP الغير متزامنة إلى الخوادم ومعالجة الاستجابات.

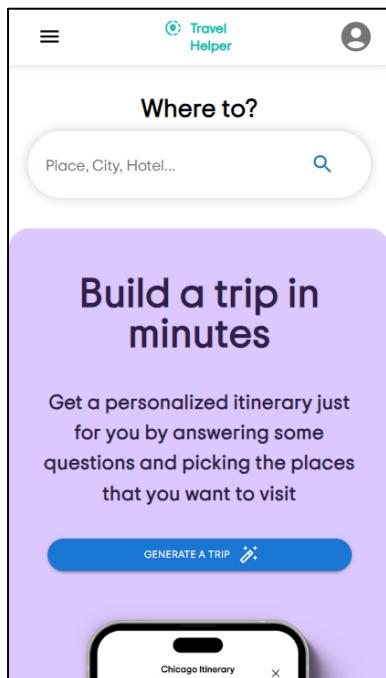
Luxon -5: مكتبة JavaScript تستخدم للعمل مع التواريخ والأوقات والمناطق الزمنية. يوفر واجهات برمجة تطبيقات قوية وبديهية للتحليل والمعالجة والتنسيق والحساب مع التواريخ والأوقات.

react-slick -6: مكتبة مكونات React شائعة تستخدم لإنشاء منزلاقات دائيرية أو عروض شرائح سريعة الاستجابة وقابلة للتخصيص في تطبيقات React. إنه مبني أعلى مكتبة slick carousel ويوفر واجهة صديقة للتفاعل لإنشاء وإدارة مكونات دائيرية.

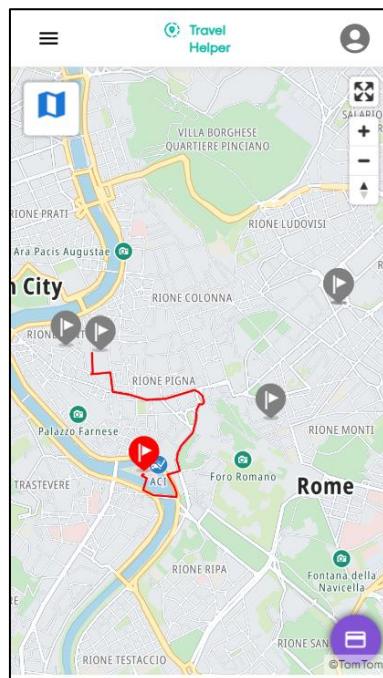
Zustand -7: مكتبة إدارة حالة (state management) خفيفة الوزن و مباشرة مصممة خصيصا لتطبيقات React. هدفها الأساسي هو تبسيط عملية التعامل مع الحالة العالمية، مما يقلل من الحاجة إلى الإعداد المفرط والتعليمات البرمجية المعварية. تشمل الخصائص البارزة لـ Zustand واجهة برمجة التطبيقات البسيطة، واستخدام hooks لتسهيل إدارة الحالة.

3-4-4- واجهات الموقع:

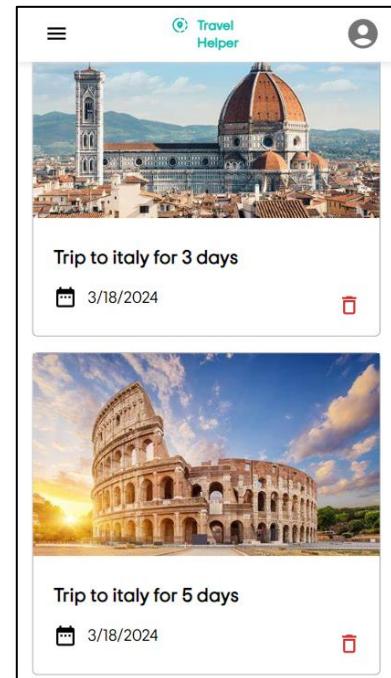
Mobile View •



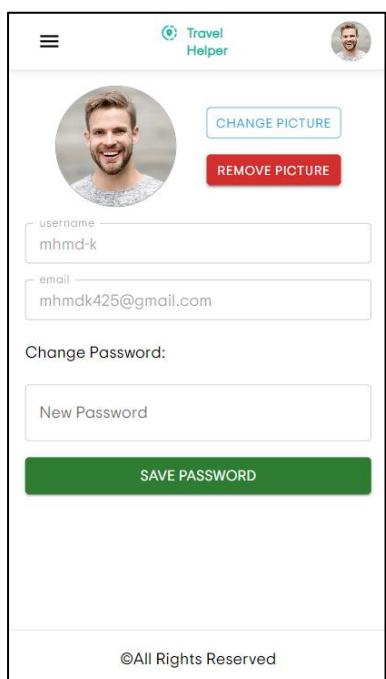
الشكل 4-7: الصفحة الرئيسية



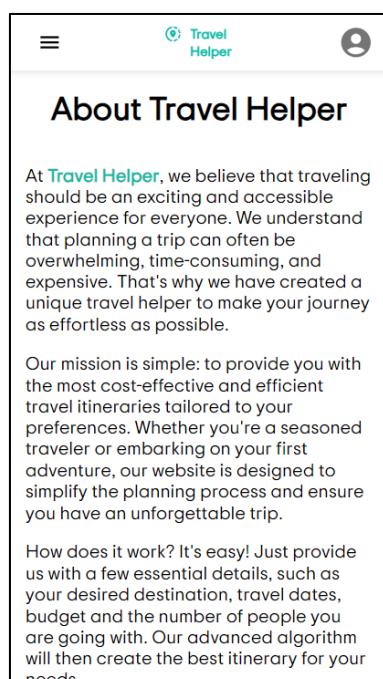
الشكل 4-6: صفحة الرحلة



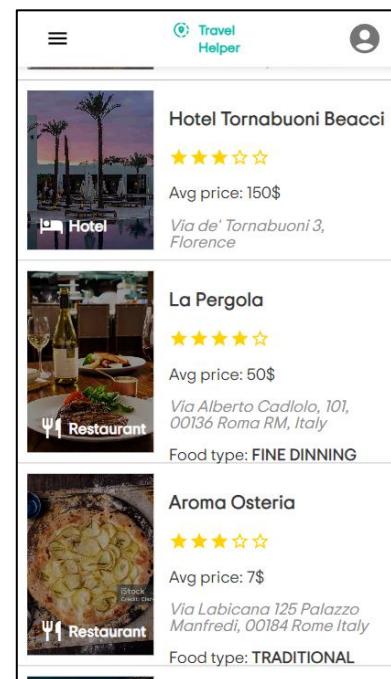
الشكل 4-5: صفحة تصفح الرحل



الشكل 4-10: صفحة الملف الشخصي



الشكل 4-9: صفحة حولنا



الشكل 4-8: صفحة البحث عن اماكن

Create Account

Username, Email, Age (0), Country (dropdown), Gender (Male), Password, Confirm Password, SUBMIT button, Already have an Account? [Login](#)

Login To Your Account

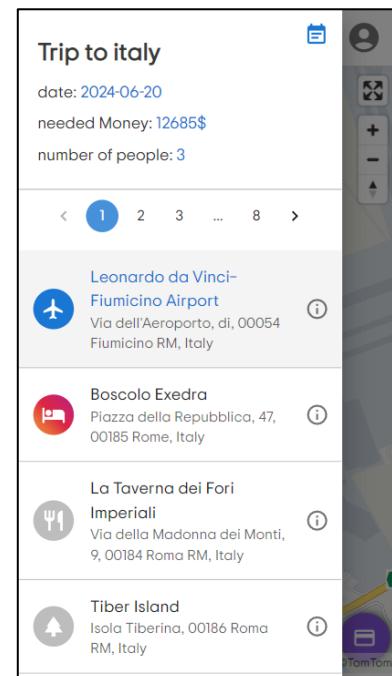
Email, Password, LOGIN button, Not a member? [Create Account](#)

Credit card

Card number * (0000 0000 0000 0000), CVV * (123), Name * (John Smith), Expiration date * (MM/YY), DONE button

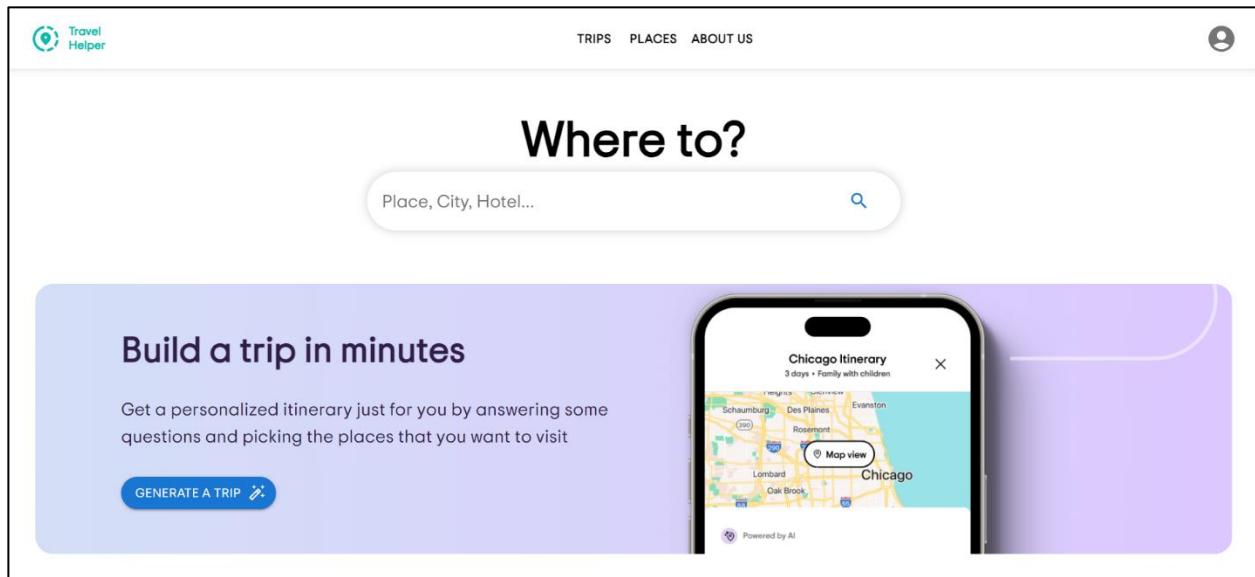
الشكل 4-12: صفحة انشاء حساب

الشكل 4-11: صفحة تسجيل الدخول

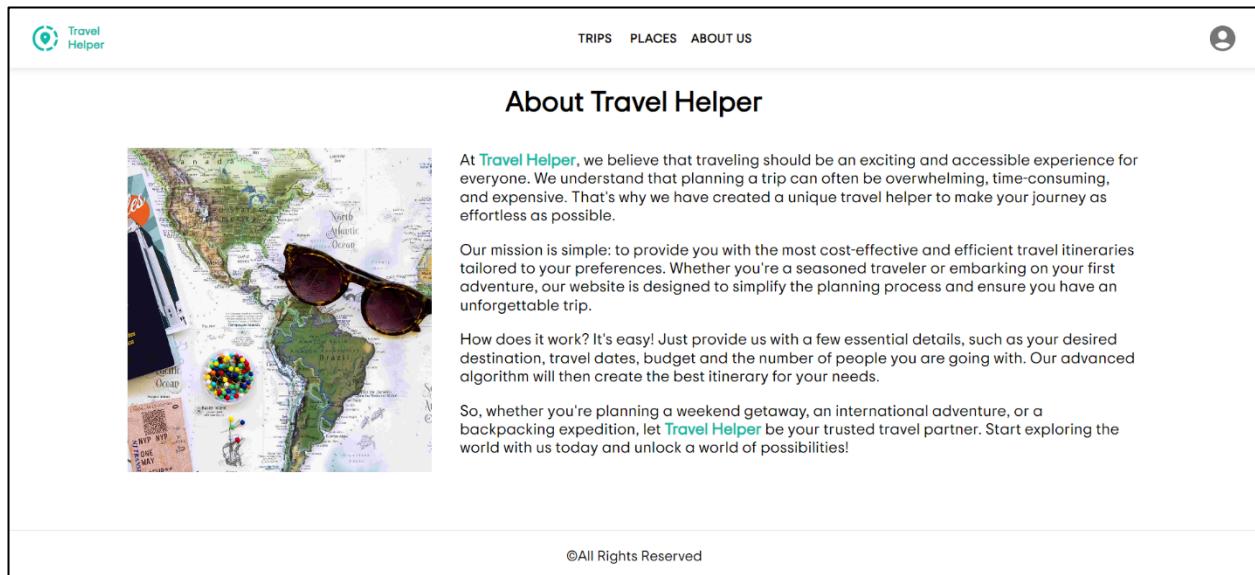


الشكل 4-13: واجهة التعامل مع الرحلة

:Desktop view •



The screenshot shows the homepage of Travel Helper. At the top, there is a navigation bar with the logo 'Travel Helper' and links for 'TRIPS', 'PLACES', and 'ABOUT US'. On the right side of the header is a user profile icon. The main title 'Where to?' is centered above a search bar with the placeholder 'Place, City, Hotel...'. Below the search bar is a large purple call-to-action box containing the text 'Build a trip in minutes' and a subtext: 'Get a personalized itinerary just for you by answering some questions and picking the places that you want to visit'. A blue button labeled 'GENERATE A TRIP' with a gear icon is at the bottom of this box. To the right of the purple box is a smartphone displaying a map of Chicago with the title 'Chicago Itinerary' and '3 days • Family with children'. The map shows the city with various travel routes and points of interest. At the bottom of the phone screen, it says 'Powered by AI'.



The screenshot shows the 'About Travel Helper' page. The top navigation bar is identical to the homepage. The main title 'About Travel Helper' is centered above the content. Below the title is a large image of a world map with various travel-related items placed on it, including a blue jacket, a pair of sunglasses, a passport, and a small globe. To the right of the image, the text reads: 'At [Travel Helper](#), we believe that traveling should be an exciting and accessible experience for everyone. We understand that planning a trip can often be overwhelming, time-consuming, and expensive. That's why we have created a unique travel helper to make your journey as effortless as possible.' Below this, another paragraph states: 'Our mission is simple: to provide you with the most cost-effective and efficient travel itineraries tailored to your preferences. Whether you're a seasoned traveler or embarking on your first adventure, our website is designed to simplify the planning process and ensure you have an unforgettable trip.' At the bottom of the page, a final paragraph reads: 'How does it work? It's easy! Just provide us with a few essential details, such as your desired destination, travel dates, budget and the number of people you are going with. Our advanced algorithm will then create the best itinerary for your needs.' At the very bottom, a small note says 'So, whether you're planning a weekend getaway, an international adventure, or a backpacking expedition, let [Travel Helper](#) be your trusted travel partner. Start exploring the world with us today and unlock a world of possibilities!' At the bottom of the page, a copyright notice reads '©All Rights Reserved'.

Travel Helper

TRIPS PLACES ABOUT US

Add some specific places

Search

Recommended to you by AI

					
+/-	+/-	-/+	+/-	+/-	-/+
RESTAURANT	RESTAURANT	HOTEL	HOTEL	NIGHT	NIGHT
Trattoria Zì? Zì?	Frito Inn	Hotel Pierre Milano	Hotel Milano Navigli	Slowly Club	Harrys Bar
city: florence	city: venice	city: milan	city: milan	city: florence	city: venice

	place	place type	city	
1	Hotel Pierre Milano	hotel	milan	trash
2	Harrys Bar	night	venice	trash

← → GENERATE TRIP 

Travel Helper

TRIPS PLACES ABOUT US

 Trip to italy for 4 days trash

5/20/2024

 Trip to italy for 3 days trash

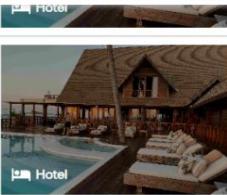
3/18/2024

 Trip to italy for 5 days trash

3/18/2024

Travel Helper

TRIPS PLACES ABOUT US



Hotel Loggiato dei Serviti
★★★★★
 Avg price: 200\$
Piazza Santissima Annunziata 3, Florence



Hotel Tornabuoni Beacci
★★★★★
 Avg price: 150\$
Via de' Tornabuoni 3, Florence



La Pergola
★★★★★
 Avg price: 50\$

Travel Helper

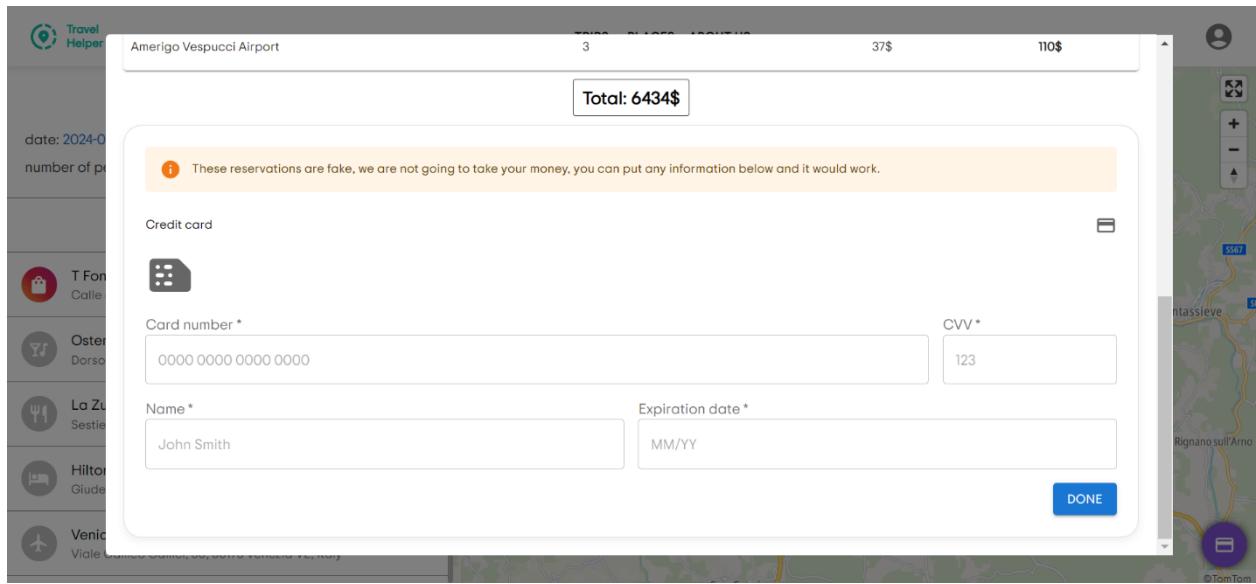
TRIPS PLACES ABOUT US

Trip Itinerary:

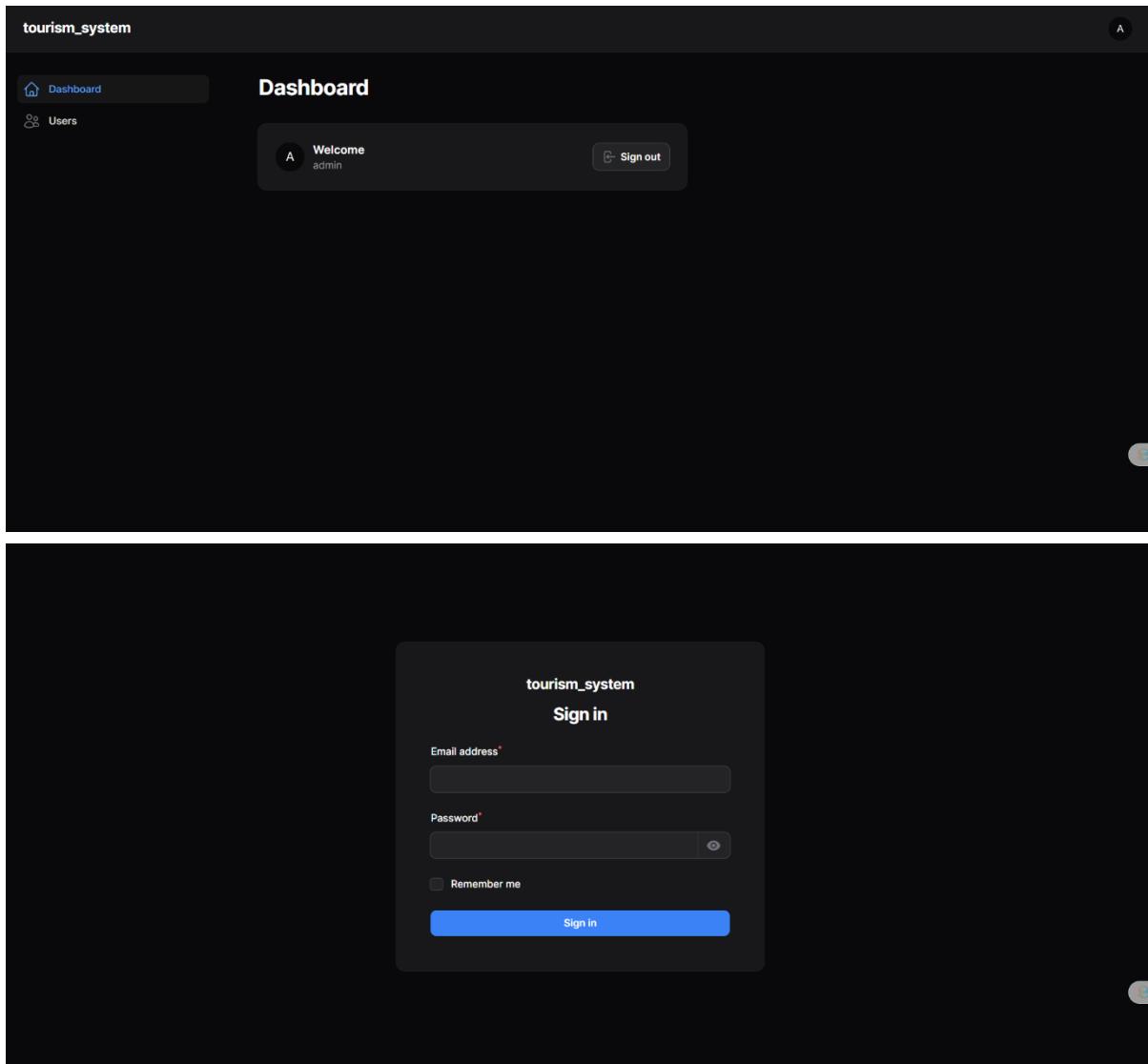
date: 2024/06/20
 number: 1

Day 1 Date: 2024/06/20 City: rome Needed Money: 1074\$

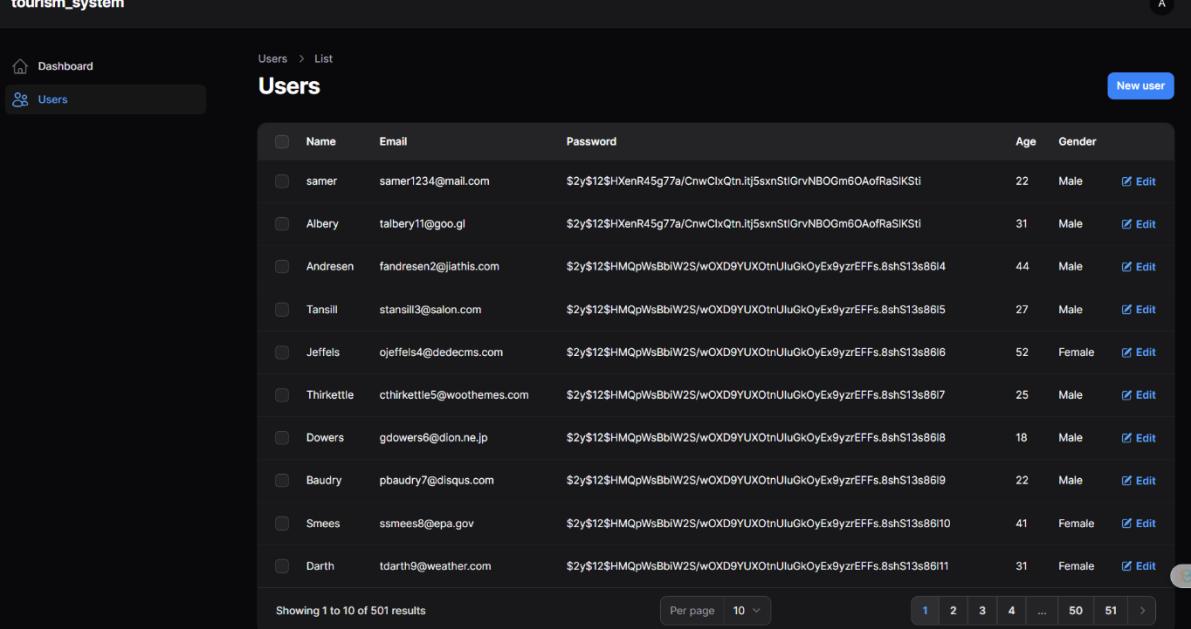
Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 10:00 AM	Leonardo da Vinci-Fiumicino Airport	2h 52min - plane	-	206\$	-	617\$
10:20 AM - 12:20 PM	Boscolo Exedra	20min - car	37\$	-	100\$	337\$
12:33 PM - 1:33 PM	La Taverna dei Fori Imperiali	13min - walking	-	-	10\$	30\$
1:46 PM - 4:46 PM	Tiber Island	13min - walking	-	-	-	0\$
4:59 PM - 8:59 PM	Piazza Navona	13min - walking	-	-	17\$	51\$
9:03 PM - 10:03 PM	433	4min - walking	-	-	13\$	39\$



4-4-4-4 : (Admin UIs) المدير واجهات (Admin UIs)



tourism_system



Dashboard > Users

Users

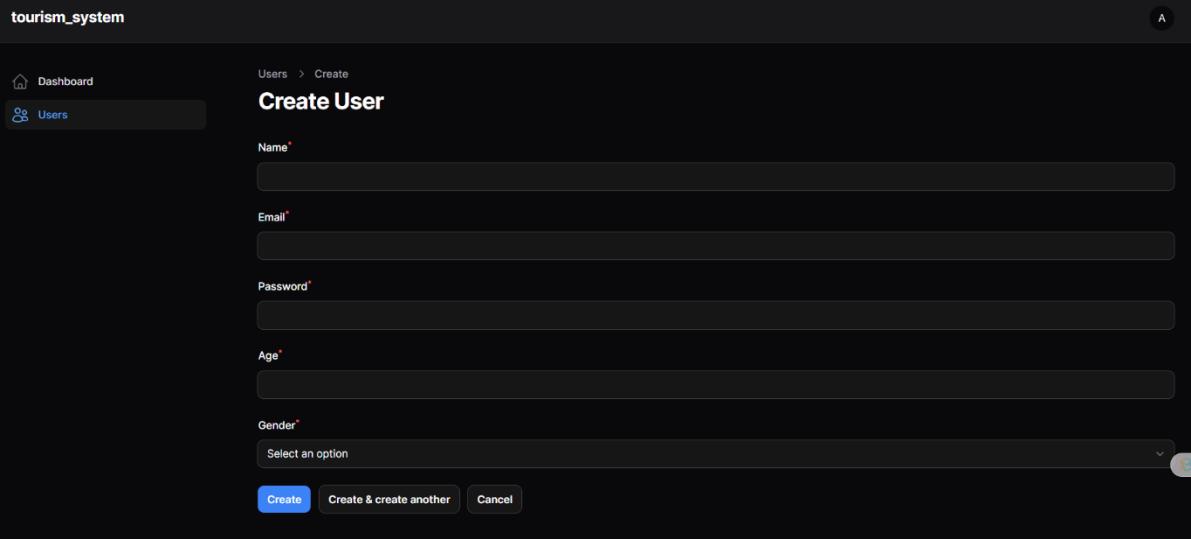
New user

	Name	Email	Password	Age	Gender	
<input type="checkbox"/>	samer	samer1234@mail.com	\$2y\$12\$HxEnR45g77a/CnwClxQtn.itj5sxnStlGrvNBOGm8OAofRaS1KStl	22	Male	Edit
<input type="checkbox"/>	Albery	talbery11@goo.gl	\$2y\$12\$HxEnR45g77a/CnwClxQtn.itj5sxnStlGrvNBOGm8OAofRaS1KStl	31	Male	Edit
<input type="checkbox"/>	Andresen	fandresen2@jathis.com	\$2y\$12\$HMQpWsBbIW25/wOXD9YUX0tnUluGkOyEx9yxrEFFs.8shS13s86i4	44	Male	Edit
<input type="checkbox"/>	Tansill	stansill3@salon.com	\$2y\$12\$HMQpWsBbIW25/wOXD9YUX0tnUluGkOyEx9yxrEFFs.8shS13s86i5	27	Male	Edit
<input type="checkbox"/>	Jeffels	ojeffels4@dedecms.com	\$2y\$12\$HMQpWsBbIW25/wOXD9YUX0tnUluGkOyEx9yxrEFFs.8shS13s86i6	52	Female	Edit
<input type="checkbox"/>	Thirkettle	cthirkettle5@woothemes.com	\$2y\$12\$HMQpWsBbIW25/wOXD9YUX0tnUluGkOyEx9yxrEFFs.8shS13s86i7	25	Male	Edit
<input type="checkbox"/>	Dowers	gdowers6@dion.ne.jp	\$2y\$12\$HMQpWsBbIW25/wOXD9YUX0tnUluGkOyEx9yxrEFFs.8shS13s86i8	18	Male	Edit
<input type="checkbox"/>	Baudry	pbaudry7@disqus.com	\$2y\$12\$HMQpWsBbIW25/wOXD9YUX0tnUluGkOyEx9yxrEFFs.8shS13s86i9	22	Male	Edit
<input type="checkbox"/>	Smee	ssmees8@epa.gov	\$2y\$12\$HMQpWsBbIW25/wOXD9YUX0tnUluGkOyEx9yxrEFFs.8shS13s86i0	41	Female	Edit
<input type="checkbox"/>	Darth	tdarth9@weather.com	\$2y\$12\$HMQpWsBbIW25/wOXD9YUX0tnUluGkOyEx9yxrEFFs.8shS13s86i11	31	Female	Edit

Showing 1 to 10 of 501 results

Per page: 10 1 2 3 4 ... 50 51 >

tourism_system



Dashboard > Users

Create User

Name*

Email*

Password*

Age*

Gender*

[Create](#) [Create & create another](#) [Cancel](#)

4-5- التطبيق والنتائج:

الآن سيتم استعراض عدة أمثلة تبين الرحلة الناتجة تبعاً لمدخلات المستخدم.

1-5-4 مثال (1):

في هذا المثال سيقوم المستخدم بإدخال ميزانية محددة (12,000\$).

مدخلات المستخدم:

input	value
From City	Damascus
To Country	Italy
Date	8/11/2024
Number of people	4
Number of days	8
Budget	12,000\$
Preferred food	traditional, dessert, fast food
Preferred places	Old, night, shopping
Specific places	<ol style="list-style-type: none">1. Palatine Hill- old place in rome2. Hotel Pierre Milano – hotel in milan3. Slowly Club- night place in florence4. Frito Inn- restaurant in venice5. Calle dei Fabbri - shopping place in venice

الرحلة الناتجة:

كما نرى في هذه الواجهة فان المسار بين كل مكائنин موضح على الخريطة التقاعدية ويمكن للمستخدم التنقل بين هذه الأماكن باستخدام قائمة الأماكن الموجودة على يسار الشاشة.

ونلاحظ انه يتم عرض يتم عرض معلومات الرحلة الرئيسية (تاريخ الرحلة، عدد الأشخاص، الميزانية، تكلفة الرحلة) في اعلى يسار الشاشة.

نلاحظ ان الرحلة الناتجة تكلفتها قريبة جدا من ميزانية المستخدم (الفرق \$11).

مسار الرحلة:

Trip Itinerary:

Day 1 Date: 2024/08/11 City: roma Needed Money: \$1,936

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 10:00 AM	Leonardo da Vinci-Fiumicino Airport	2h 52min - plane	-	\$206	-	\$822
10:18 AM - 12:18 PM	Hotel Ponte Sisto	18min - car	\$34	-	\$200	\$834
12:27 PM - 1:27 PM	Kebab	9min - walking	-	-	\$13	\$52
1:45 PM - 6:45 PM	Palatine Hill	18min - walking	-	-	\$27	\$108
7:02 PM - 8:02 PM	Nannarella	17min - walking	-	-	\$30	\$120
8:08 PM	Hotel Ponte Sisto	6min - walking	-	-	\$200	\$0

Day 2 Date: 2024-08-12 City: roma Needed Money: \$1,112

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 11:00 AM	Hotel Ponte Sisto	-	-	-	\$200	\$800
11:07 AM - 12:07 PM	Frigidarium	7min - walking	-	-	\$10	\$40
12:21 PM - 4:21 PM	Via Condotti	14min - walking	-	-	-	\$0
4:32 PM - 7:32 PM	Sharivari	11min - walking	-	-	\$60	\$240
7:47 PM - 8:47 PM	Ai Tre Scalini	15min - walking	-	-	\$8	\$32
9:06 PM	Hotel Ponte Sisto	19min - walking	-	-	\$200	\$0

Day 3 Date: 2024-08-13 City: milan Needed Money: \$1,166

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 10:00 AM	Milan Malpensa Airport	36min - plane	-	\$43	-	\$172
10:35 AM - 12:35 PM	Hotel Pierre Milano	35min - car	\$64	-	\$150	\$664
12:54 PM - 1:54 PM	Trattoria Toscana	19min - walking	-	-	\$15	\$60
2:10 PM - 6:10 PM	Santa Maria delle Grazie	16min - walking	-	-	\$17	\$68
6:44 PM - 9:44 PM	The Lounge Bar:	34min - walking	-	-	\$34	\$136
9:54 PM - 10:54 PM	Ratana	10min - walking	-	-	\$15	\$60
10:57 PM	Hotel Pierre Milano	3min - car	\$6	-	\$150	\$6

Travel

Day 4 Date: 2024-08-14 City: milan Needed Money: \$941

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 11:00 AM	Hotel Pierre Milano	-	-	-	\$150	\$600
11:23 AM - 12:23 PM	Gli Orti Del Belvedere	23min - walking	-	-	\$12	\$48
12:42 PM - 3:42 PM	Porta Ticinese	19min - walking	-	-	-	\$0
3:51 PM - 7:51 PM	Navigli Canals	9min - walking	-	-	\$27	\$108
8:25 PM - 9:25 PM	Trottoria Milanese	34min - car	\$63	-	\$15	\$123
9:59 PM	Hotel Pierre Milano	34min - car	\$62	-	\$150	\$62

Travel

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 10:00 AM	Naples International Airport	50min - plane	-	\$60	-	\$237
10:06 AM - 12:06 PM	Renaissance Naples Hotel Mediterraneo	6min - car	\$10	-	\$240	\$970
12:08 PM - 1:08 PM	McDonald's 2	2min - walking	-	-	\$12	\$48
1:19 PM - 3:19 PM	San Domenico Maggiore	11min - walking	-	-	\$20	\$80
3:31 PM - 6:31 PM	Quartieri Spagnoli	12min - walking	-	-	-	\$0
6:33 PM - 8:33 PM	Quartieri Spagnoli	2min - walking	-	-	\$35	\$140
8:37 PM - 9:37 PM	Trattoria da Nennella	4min - walking	-	-	\$9	\$36
9:42 PM	Renaissance Naples Hotel Mediterraneo	5min - walking	-	-	\$240	\$0

Travel

Day 6 Date: 2024-08-16 City: napoli Needed Money: \$1,168

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 11:00 AM	Renaissance Naples Hotel Mediterraneo	-	-	-	\$240	\$960
11:06 AM - 12:06 PM	Pizzeria da Attilio	6min - walking	-	-	\$10	\$40
12:18 PM - 4:18 PM	Via San Gregorio Armeno	12min - walking	-	-	-	\$0
4:35 PM - 7:35 PM	Molo Beverello	17min - walking	-	-	\$34	\$136
7:53 PM - 8:53 PM	Pizzeria Di Matteo	18min - walking	-	-	\$8	\$32
9:09 PM	Renaissance Naples Hotel Mediterraneo	16min - walking	-	-	\$240	\$0

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 10:00 AM	Amerigo Vespucci Airport	31min - plane	-	\$37	-	\$147
10:05 AM - 12:05 PM	Hotel Degli Artisti	5min - car	\$8	-	\$240	\$968
12:16 PM - 1:16 PM	Il Santo Graal	11min - walking	-	-	\$10	\$40
1:20 PM - 3:20 PM	Santa Maria del Carmine	4min - walking	-	-	\$23	\$92
3:35 PM - 6:35 PM	The Mall Firenze	15min - walking	-	-	-	\$0
6:45 PM - 9:45 PM	Slowly Club	10min - car	\$19	-	\$54	\$235
9:56 PM - 10:56 PM	Konnubio	11min - car	\$20	-	\$10	\$60
11:07 PM	Hotel Degli Artisti	11min - walking	-	-	\$240	\$0

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 11:00 AM	Hotel Degli Artisti	-	-	-	-	\$0
1:55 PM - 3:55 PM	Hotel Santa Chiara	2h 55min - car	\$326	-	\$300	\$1,526
4:07 PM - 5:07 PM	Frito Inn	12min - walking	-	-	\$16	\$64
5:23 PM - 8:23 PM	Calle dei Fabbri	16min - walking	-	-	-	\$0
8:29 PM - 11:29 PM	The Gritti Palace Terrace	6min - walking	-	-	\$30	\$120
11:44 PM - 12:44 AM	La Zucca	15min - walking	-	-	\$8	\$32
12:53 AM - 2:53 AM	Hotel Santa Chiara	9min - walking	-	-	\$300	\$0
3:00 AM	Venice Marco Polo Airport	7min - car	\$13	\$220	-	\$893

كما نرى فإن الجدول الزمني للرحلة شامل لكل المعلومات التي تهم المستخدم أثناء رحلته، حيث انه يتم عرض أوقات التواجد في كل مكان، الزمن التقديرى المستغرق في المواصلات، وسيلة المواصلات المقترنة للوصول الى كل مكان، التكلفة الفردية والتكلفة الكلية لكل مكان.

ونلاحظ وجود تنوع بأنواع الأماكن في اليوم الأول من الرحلة سيزور المستخدم مكان اثري (palatine hill) وفي اليوم التالي سيزور مكان سوق ومكان ليلي (عدم تكرار أنواع الأماكن).

ونلاحظ أن جميع الأماكن التي اضافها المستخدم موجودة ضمن رحلته.

الحجوزات الازمة:

hotel	address	number of people	price per person	total amount
Hotel Ponte Sisto	Via dei Pettinari, 64, 00186 Rome, Italy	4	200\$	800\$
Hotel Pierre Milano	Via De Amicis, 32	4	150\$	600\$
Renaissance Naples Hotel Mediterraneo	Via Ponte di Tappia, 25, 80133 Naples NA, Italy	4	240\$	960\$
Hotel Degli Artisti	Via del Giglio 87, Florence	4	240\$	960\$
Hotel Santa Chiara	Santa Croce, 30135 Venezia VE, Italy	4	300\$	1200\$
airport	number of tickets	ticket price	total amount	
Leonardo da Vinci-Fiumicino Airport	4	206\$	822\$	
Milan Malpensa Airport	4	43\$	172\$	
Naples International Airport	4	60\$	237\$	
Amerigo Vespucci Airport	4	37\$	147\$	

Total: 5898\$

مثال (2) - 2-5-4

مدخلات المستخدم هي نفسها المدخلات في المثال رقم (1) ولكن هنا المستخدم يريد أرخص رحلة ممكنة.

الرحلة الناتجة:

date	needed Money	num of people
2024-08-11	\$6,449	4

Trip to Italy

date	needed Money	num of people
2024-08-11	\$6,449	4

needed Money: \$6,449

num of people: 4

1 2 3 ... 8 >

Leonardo da Vinci-Fiumicino Airport Via dell'Aeroporto, 00054 Fiumicino RM, Italy	(i)
The St. Regis Rome Via Vittorio Emanuele Orlando, 3, 00185 Rome, Italy	(i)
Roscioli Caffè Pasticceria Piazza Benedetto Cairoli, 16, 00186 Roma RM, Italy	(i)
Palatine Hill Via di San Gregorio, 00186 Roma RM, Italy	(i)
Imago Piazza della Trinità Dei Monti 6, Hassler Roma Hotel - 00187 Rome	(i)
The St. Regis Rome Via Vittorio Emanuele Orlando, 3, 00185 Rome, Italy	(i)

نلاحظ الفرق الكبير في تكلفة الرحلة وذلك لأن المستخدم طلب أرخص رحلة ممكنة في هذا المثال.

مسار الرحلة:

Trip Itinerary:

Day 1 Date: 2024/08/11 City: roma Needed Money: \$1,199

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 10:00 AM	Leonardo da Vinci-Fiumicino Airport	2h 52min - plane	-	\$206	-	\$822
10:20 AM - 12:20 PM	The St. Regis Rome	20min - car	\$37	-	\$50	\$237
12:41 PM - 1:41 PM	Roscioli Caffè Pasticceria	21min - walking	-	-	\$4	\$16
1:52 PM - 6:52 PM	Palatine Hill	11min - walking	-	-	\$27	\$108
7:14 PM - 8:14 PM	Imago	22min - walking	-	-	\$4	\$16
8:26 PM	The St. Regis Rome	12min - walking	-	-	\$50	\$0

Day 2 Date: 2024-08-12 City: roma Needed Money: \$480

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 11:00 AM	The St. Regis Rome	-	-	-	\$50	\$200
11:11 AM - 12:11 PM	McDonald's	11min - walking	-	-	\$5	\$20
12:14 PM - 4:14 PM	Burberry	3min - walking	-	-	-	\$0
4:27 PM - 7:27 PM	Sharivari	13min - walking	-	-	\$60	\$240
7:40 PM - 8:40 PM	Tonnarello	13min - walking	-	-	\$5	\$20
9:06 PM	The St. Regis Rome	26min - walking	-	-	\$50	\$0

Day 3 Date: 2024-08-13 City: milan Needed Money: \$948

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 10:00 AM	Milan Malpensa Airport	36min - plane	-	\$43	-	\$172
10:35 AM - 12:35 PM	Hotel Pierre Milano	35min - car	\$64	-	\$150	\$664
12:45 PM - 1:45 PM	Langosteria Bistrot	10min - walking	-	-	\$3	\$12
1:58 PM - 5:58 PM	The Columns of San Lorenzo (Colonne di San Lorenzo)	13min - walking	-	-	\$8	\$32
6:07 PM - 9:07 PM	Hole Rabbit Cafè	9min - walking	-	-	\$12	\$48
9:35 PM - 10:35 PM	Antica Trattoria della Pesa	28min - walking	-	-	\$5	\$20
11:06 PM	Hotel Pierre Milano	31min - walking	-	-	\$150	\$0

Travel

Day 4 Date: 2024-08-14 City: milan Needed Money: \$738

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 11:00 AM	Hotel Pierre Milano	-	-	\$150	\$600	\$600
11:22 AM - 12:22 PM	Nobu Milano	22min - walking	-	\$4	\$16	\$16
12:26 PM - 4:26 PM	Via della Spiga	4min - walking	-	-	\$0	\$0
4:58 PM - 7:58 PM	Rock & Roll Milano	32min - walking	-	\$23	\$92	\$92
8:01 PM - 9:01 PM	Ristorante Da Giacomo	3min - car	\$6	-	\$6	\$30
9:15 PM	Hotel Pierre Milano	14min - walking	-	\$150	\$0	\$0

Day 5 Date: 2024-08-15 City: napoli Needed Money: \$525

Travel

Day 5 Date: 2024-08-15 City: napoli Needed Money: \$525

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 10:00 AM	Naples International Airport	50min - plane	-	\$60	-	\$237
10:06 AM - 12:06 PM	Grand Hotel Vesuvio	6min - car	\$10	-	\$50	\$210
12:11 PM - 1:11 PM	McDonald's	5min - walking	-	-	\$5	\$20
1:42 PM - 4:42 PM	San Lorenzo Maggiore	31min - walking	-	-	-	\$0
5:01 PM - 9:01 PM	Mercato di Porta Nolana	19min - walking	-	-	-	\$0
9:12 PM - 10:12 PM	Trattoria Da Nennella	11min - car	\$20	-	\$4	\$36
10:24 PM	Grand Hotel Vesuvio	12min - car	\$22	-	\$50	\$22

Travel

Day 6 Date: 2024-08-16 City: napoli Needed Money: \$380

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 11:00 AM	Grand Hotel Vesuvio	-	-	-	\$50	\$200
11:05 AM - 12:05 PM	Pasticceria Calemma Dolci Momenti	5min - walking	-	-	\$6	\$24
12:22 PM - 4:22 PM	Via San Gregorio Armeno	17min - walking	-	-	-	\$0
4:39 PM - 7:39 PM	Molo Beverello	17min - walking	-	-	\$34	\$136
8:04 PM - 9:04 PM	Ristorante A Figlia d'O Luciano	25min - walking	-	-	\$5	\$20
9:31 PM	Grand Hotel Vesuvio	27min - walking	-	-	\$50	\$0

00187 Rome

Travel

TRIPS PLACES ABOUT US

Day 7 Date: 2024-08-17 City: florence Needed Money: \$680

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 10:00 AM	Amerigo Vespucci Airport	31min - plane	-	\$37	-	\$147
10:06 AM - 12:06 PM	Hotel Brunelleschi	6min - car	\$10	-	\$50	\$210
12:16 PM - 1:16 PM	Il Santo Bevitore	10min - walking	-	-	\$3	\$12
1:25 PM - 5:25 PM	Orsanmichele	9min - walking	-	-	\$10	\$40
5:36 PM - 8:36 PM	Slowly Club	11min - car	\$20	-	\$54	\$236
8:47 PM - 9:47 PM	Trattoria Z♦♦? Z♦♦?	11min - car	\$19	-	\$4	\$35
9:54 PM	Hotel Brunelleschi	7min - walking	-	-	\$50	\$0

00187 Rome



Travel

TRIPS PLACES ABOUT US

Day 8 Date: 2024-08-18 City: venice Needed Money: \$1,499

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 11:00 AM	Hotel Brunelleschi	-	-	-	-	\$0
1:55 PM - 3:55 PM	Hotel Iris	2h 55min - car	\$326	-	\$50	\$326
4:07 PM - 5:07 PM	Frito Inn	12min - walking	-	-	\$16	\$64
5:23 PM - 8:23 PM	Calle dei Fabbri	16min - walking	-	-	-	\$0
8:27 PM - 12:27 AM	Rialto Bridge	4min - walking	-	-	-	\$0
12:32 AM - 1:32 AM	Osteria alle Testiere	5min - walking	-	-	\$4	\$16
1:46 AM - 3:46 AM	Hotel Iris	14min - walking	-	-	\$50	\$0
3:53 AM	Venice Marco Polo Airport	7min - car	\$13	\$220	-	\$893



كما نلاحظ فإن جميع الأماكن التي اضافها المستخدم موجودة ضمن رحلته.

4-5-3- مثال (3):

في هذا المثال سيقوم المستخدم بإدخال ميزانية محددة ولكنها غير منطقية بالمقارنة مع عدد أيام الرحلة وعدد الأشخاص، وعند حدوث هذه الحالة فإنه يتم عرض رسالة للمستخدم بان الميزانية غير كافية.

مدخلات المستخدم:

input	value
From City	Cairo
To Country	Italy
Date	9/5/2024
Number of people	8
Number of days	5
Budget	2,500\$
Preferred food	Fine dining, dessert, fast food
Preferred places	Old, natural, night
Specific places	<ol style="list-style-type: none"> 1. Frito Inn-restaurant in Venice. 2. Hotel Milano Navigli-hotel in Milan. 3. Harrys bar-night place in Venice. 4. Slowly club-night place in Florence.

الرحلة الناتجة:

رسالة التنبية

Trip to Italy

date: 2024-09-05

budget: \$2,500

needed Money: \$9,399

num of people: 8

TRIPS PLACES ABOUT US

Leonardo da Vinci-Fiumicino Airport

The St. Regis Rome

Five Guys

Basilica of Santa Maria Maggiore

Botanical Garden of Rome

Roscioli Caffè Pasticceria

Map showing a route from Fiumicino to Rome, with a red line indicating the path. A red box highlights the message: "Your budget is so low, The cheapest trip costs \$9,399".

نلاحظ وجود فرق كبير بين تكلفة الرحلة (9,399\$) وميزانية المستخدم (2,500\$) بسبب مدخلات المستخدم الغير منطقية وعند حدوث هذا الامر يتم عرض رسالة تنبية للمستخدم بان ميزانيته غير كافية.

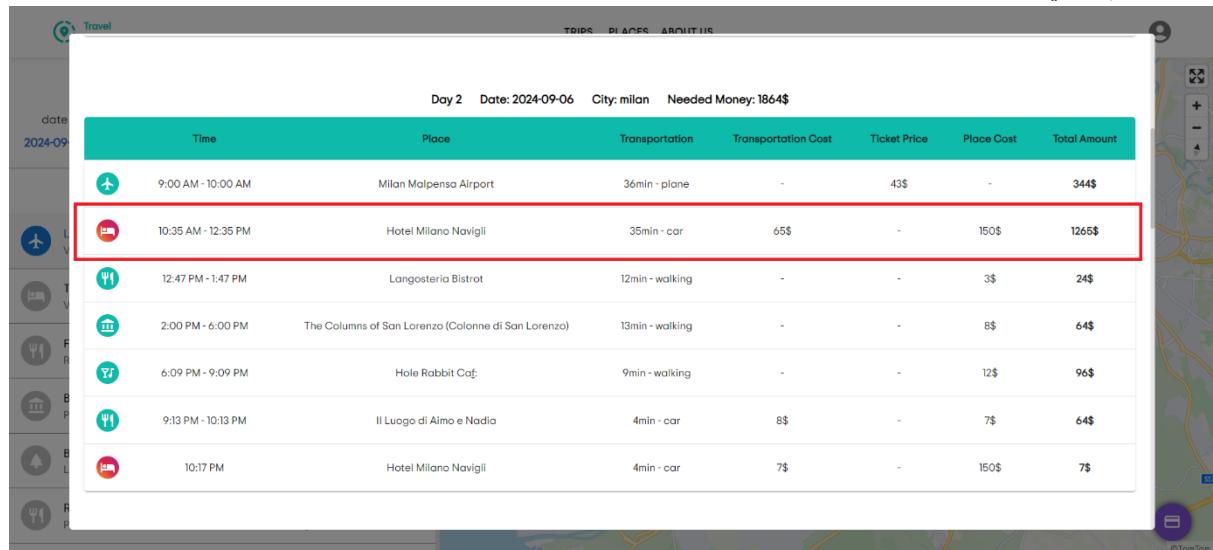
مسار اليوم الأول:

Trip Itinerary:

Day 1 Date: 2024/09/05 City: rome Needed Money: 2086\$

Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 10:00 AM	Leonardo da Vinci-Fiumicino Airport	2h 41min - plane	-	193\$	-	1637\$
10:20 AM - 12:20 PM	The St. Regis Rome	20min - car	37\$	-	50\$	437\$
12:29 PM - 1:29 PM	Five Guys	9min - walking	-	-	5\$	40\$
1:42 PM - 4:42 PM	Basilica of Santa Maria Maggiore	13min - walking	-	-	5\$	40\$
5:18 PM - 9:18 PM	Botanical Garden of Rome	36min - walking	-	-	-	0\$
9:34 PM - 10:34 PM	Roscioli Caffè Pasticceria	16min - walking	-	-	4\$	32\$
10:55 PM	The St. Regis Rome	21min - walking	-	-	50\$	0\$

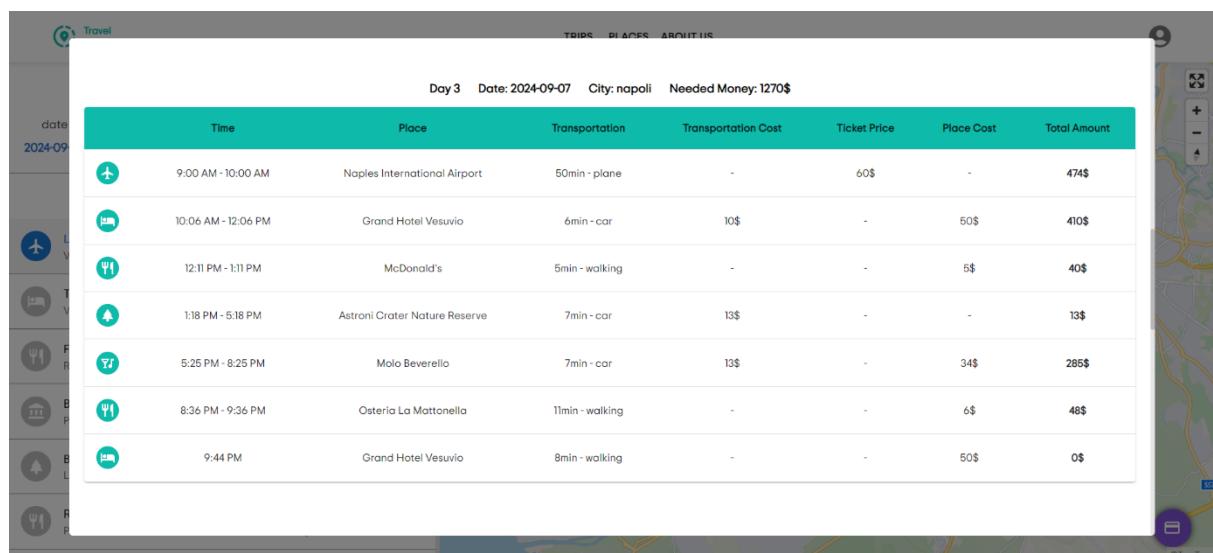
مسار اليوم الثاني:



Day 2 Date: 2024-09-06 City: milan Needed Money: 1864\$							
Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount	
9:00 AM - 10:00 AM	Milan Malpensa Airport	36min - plane	-	43\$	-	344\$	
10:35 AM - 12:35 PM	Hotel Milano Navigli	35min - car	65\$	-	150\$	1265\$	
12:47 PM - 1:47 PM	Langosteria Bistrot	12min - walking	-	-	3\$	24\$	
2:00 PM - 6:00 PM	The Columns of San Lorenzo (Colonne di San Lorenzo)	13min - walking	-	-	8\$	64\$	
6:09 PM - 9:09 PM	Hole Rabbit Cafè	9min - walking	-	-	12\$	96\$	
9:13 PM - 10:13 PM	Il Luogo di Aimo e Nadia	4min - car	8\$	-	7\$	64\$	
10:17 PM	Hotel Milano Navigli	4min - car	7\$	-	150\$	7\$	

نلاحظ وجود الفندق الذي ادخله المستخدم في مدينة ميلان ضمن مسار اليوم الثاني.

مسار اليوم الثالث:



Day 3 Date: 2024-09-07 City: napoli Needed Money: 1270\$							
Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount	
9:00 AM - 10:00 AM	Naples International Airport	50min - plane	-	60\$	-	474\$	
10:06 AM - 12:06 PM	Grand Hotel Vesuvio	6min - car	10\$	-	50\$	410\$	
12:11 PM - 1:11 PM	McDonald's	5min - walking	-	-	5\$	40\$	
1:18 PM - 5:18 PM	Astroni Crater Nature Reserve	7min - car	13\$	-	-	13\$	
5:25 PM - 8:25 PM	Molo Beverello	7min - car	13\$	-	34\$	285\$	
8:36 PM - 9:36 PM	Osteria La Mattonella	11min - walking	-	-	6\$	46\$	
9:44 PM	Grand Hotel Vesuvio	8min - walking	-	-	50\$	0\$	

مسار اليوم الرابع:



Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 10:00 AM	Amerigo Vespucci Airport	31min - plane	-	37\$	-	293\$
10:06 AM - 12:06 PM	Hotel Brunelleschi	6min - car	10\$	-	50\$	410\$
12:13 PM - 1:13 PM	La Buchetta Food & Wine	7min - walking	-	-	4\$	32\$
1:19 PM - 5:19 PM	Orsanmichele	6min - walking	-	-	10\$	80\$
5:30 PM - 8:30 PM	Slowly Club	11min - car	20\$	-	54\$	462\$
8:41 PM - 9:41 PM	Il Santo Bevitore	11min - car	21\$	-	3\$	46\$
9:51 PM	Hotel Brunelleschi	10min - walking	-	-	50\$	0\$

نلاحظ وجود المكان (Slowly Club) الذي ادخله المستخدم في مدينة فلورنس في مسار اليوم الرابع.

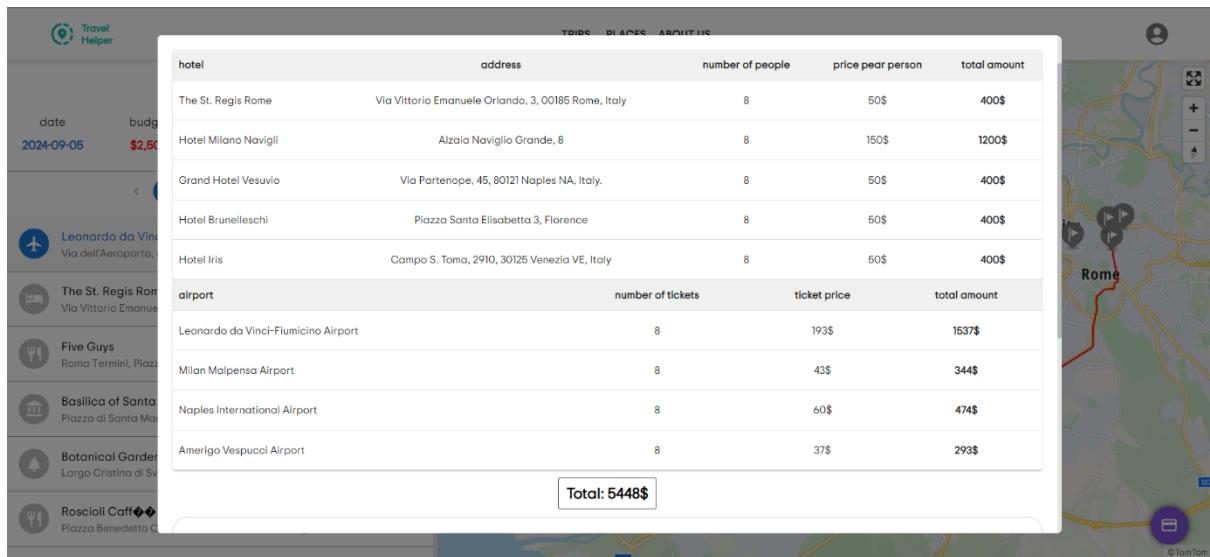
مسار اليوم الخامس والأخير:



Time	Place	Transportation	Transportation Cost	Ticket Price	Place Cost	Total Amount
9:00 AM - 11:00 AM	Hotel Brunelleschi	-	-	-	-	0\$
1:55 PM - 3:55 PM	Hotel Iris	2h 55min - car	326\$	-	50\$	726\$
4:07 PM - 5:07 PM	Frito Inn	12min - walking	-	-	16\$	128\$
5:10 PM - 7:10 PM	Parco San Giacomo dell'Orio	3min - walking	-	-	-	0\$
7:14 PM - 10:14 PM	Ghetto Ebraico - Comunità Ebraica di Venezia	4min - walking	-	-	12\$	96\$
10:34 PM - 12:34 AM	Harrys Bar	20min - walking	-	-	20\$	160\$
12:46 AM - 1:46 AM	GROM	12min - walking	-	-	3\$	24\$
1:50 AM - 3:50 AM	Hotel Iris	4min - walking	-	-	50\$	0\$
3:57 AM	Venice Marco Polo Airport	7min - car	13\$	215\$	-	1733\$

نلاحظ وجود المكانين (Frito Inn, Harrys Bar) اللذين ادخلهما المستخدم في مدينة فينيس في مسار اليوم الخامس.

الحجوزات اللازمة للرحلة:



The screenshot shows a travel planning application interface. On the left, there's a sidebar with icons for Leonardo da Vinci Airport, The St. Regis Rome, Five Guys, Basilica di Santa Maria, Botanical Garden, and Roscioli Caffè. The main area has tabs for 'TRIPS', 'PLACES', and 'ARCHITECTURE'. A table for 'hotels' lists five options with details like address, number of people, price per person, and total amount. Another table for 'airports' lists four options with details like number of tickets, ticket price, and total amount. A total amount of 5448\$ is displayed. To the right is a map of Rome with a red route line and a legend.

hotel	address	number of people	price per person	total amount
The St. Regis Rome	Via Vittorio Emanuele Orlando, 3, 00185 Rome, Italy	8	50\$	400\$
Hotel Milano Navigli	Alzaia Naviglio Grande, 8	8	150\$	1200\$
Grand Hotel Vesuvio	Via Partenope, 45, 80121 Naples NA, Italy.	8	50\$	400\$
Hotel Brunelleschi	Piazza Santa Elisabetta 3, Florence	8	50\$	400\$
Hotel Iris	Campo S. Toma, 2910, 30125 Venezia VE, Italy	8	50\$	400\$

airport	number of tickets	ticket price	total amount
Leonardo da Vinci-Fiumicino Airport	8	193\$	1537\$
Milan Malpensa Airport	8	43\$	344\$
Naples International Airport	8	60\$	474\$
Amerigo Vespucci Airport	8	37\$	293\$

Total: 5448\$

هذه هي الخطوات التي تم اتباعها أثناء عملية التطبيق العملي حيث انه تم جمع بيانات عم 685 مكان سياحي في إيطاليا وبعد ذلك تم تطوير موقع ويب وتم استخدام مكتبة React لتطوير الواجهات الامامية وإطار عمل Laravel لتطوير الواجهات الخلفية. في النهاية تم اختبار الموقع عرض نتائج الاختبارات المختلفة في عدة امثلة.

الخاتمة

لقد تم تطوير نظام مساعد للسياحة كامل يوفر للمستخدم الميزات التالية: مسار كل يوم في الرحلة، معلومات عن كل مكان والمواصلات وأوقات زيارة كل مكان وسعر دخوله، خريطة تفاعلية، حجز تذاكر طيران وحجوزات فندقية، والعديد من الميزات الأخرى.

من أكبر التحديات التي واجهتنا أثناء التطوير هي كمية البيانات القليلة المتاحة، حيث تبين أن بناء نظام مساعد للسياحة يتطلب وجود كمية بيانات ضخمة عن الأماكن المختلفة حول العالم.

على الرغم من أن النظام يحتوي على العديد من الميزات إلا أنه لا يوجد نظام كامل وهناك دائماً مكان للتحسين، من التطويرات المستقبلية الممكن القيام بها:

- جمع بيانات لبقية الدول لكي يصبح النظام شامل لبقية العالم وليس فقط إيطاليا.
- الحصول على API's من شركات طيران وبالتالي يصبح بإمكان المستخدم الاستفادة من العروض المختلفة ويصبح بالإمكان الحجز بعدة درجات (درجة اقتصادية، درجة الاعمال، والدرجة الأولى).
- الحصول على API's لحجوزات الفنادق وبالتالي يصبح بإمكان المستخدم حجز غرفة لشخص أو لشخصين أو حجز جناح (حالياً كل فندق يملك سعر ثابت للشخص الواحد).
- الحصول على لوائح الوجبات الأسعار من المطاعم المختلفة بحيث يصبح لـ المستخدم قائمة الوجبات واسعارها لكل مطعم (حالياً يوجد سعر وسطي ثابت لكل مطعم).
- إضافة قسم مقالات ومنشورات لكي يستفيد المستخدمين من تجارب بعضهم.

المراجع

- [1] Badouch, M., & Boutaoune, M. (2023, April). Personalized Travel Recommendation Systems: A Study of Machine Learning Approaches in Tourism. *Journal of Artificial Intelligence Machine Learning and Neural Network*. Retrieved from <https://doi.org/10.55529/jaimlnn.33.35.45>
- [2] Bhargava , A. (2016). *Grokking Algorithms: An Illustrated Guide for Programmers and Other Curious People*. Manning Publications.
- [3] Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. O'Reilly Media, Inc.
- [4] Liu, L.-s., Lin, J.-f., Yao, J.-x., He, D.-w., Zheng, J.-s., Huang, J., & Shi, P. (2021, February). Path Planning for Smart Car Based on Dijkstra Algorithm and Dynamic Window Approach. *Wireless Communications and Mobile Computing*. Retrieved from <https://doi.org/10.1155/2021/8881684>
- [5] (n.d.). Retrieved from Trip advisor: <https://www.tripadvisor.com/AITripBuilder>
- [6] (n.d.). Retrieved from Lonely planet: <https://www.lonelyplanet.com/>
- [7] Otwell, T. (2024, May). *Laravel – The PHP Framework For Web Artisans*. Retrieved from Laravel: <https://laravel.com/>
- [8] Shrivastava, S. (2023). Retrieved from Kaggle: <https://www.kaggle.com/code/djangomustang/travel-recommendation-system>
- [9] Staudemeyer, R. C., & Morris, E. R. (2019). *Understanding LSTM -- a tutorial into Long Short-Term Memory Recurrent Neural Networks*. Retrieved from <https://arxiv.org/pdf/1909.09586>
- [10] Stauffer, M. (2019). *Laravel: Up & Running, 3rd Edition*. O'Reilly Media, Inc.
- [11] Walke, J. (2024, April). *React*. Retrieved from react.dev: <https://react.dev/>

