

# تقرير التحليل المعماري واستراتيجية التنفيذ لمنصة في ضوء Manus Delivery Ways: مراجعة خطة والنماذج المرنة "Uber" معايير

## الملخص التنفيذي والسياق الاستراتيجي 1.

### تحول النموذج الهندسي: من الكود اليدوي إلى التوليد المعزز بالذكاء الاصطناعي 1.1

يشهد المشهد الهندسي للبرمجيات حالياً تحولاً زلزالياً، يتميز بالانتقال من كتابة الأكواد البرمجية يدوياً بشكل كامل إلى توليد البنية التحتية كما هو محدد في التذكرة ، DW ("Delivery Ways") المعمارية بمساعدة وكلاء الذكاء الاصطناعي المتقدمين. يمثل مشروع وهو وكيل ذكاء اصطناعي - "Manus" مثلاً جوهرياً لهذا النموذج الحديث. من خلال الاستفادة من ، ANALYSIS-PLAN-001، (Prototype) عالي القدرة مصمم لتنفيذ المهام المعقّدة وتوليد الأكواد<sup>1</sup> - يهدف المشروع إلى تسريع مساره من مرحلة النموذج الأولي "Uber" إلى منصة جاهزة للإنتاج بمستوى يضاهي

يقدم مجموعة فريدة من التحديات المتعلقة بحوكمة الكود، والنزاهة Manus ومع ذلك، فإن استخدام وكلاء الذكاء الاصطناعي مثل (Refactoring)، على عكس إعادة الهيكلة التي يقوم بها البشر. (Legacy Systems)، على حساب (Localized Optimization) غالباً ما تعطي الخطط التي يولدها الذكاء الاصطناعي الأولوية للسرعة والتحسين المحلي المتضمنة في Manus إن تحليل خطة. (Hard Constraints) "الاتساق النظامي الشامل، ما لم يتم تقييدها بصرامة بـ "قيود صلبة ليس مجرد مراجعة للكود؛ بل هو تدقيق معماري Delivery Ways - Project Development Plan.zip الملف المضغوط لمخرجات وكيل مستقل مقابل المعايير الصارمة لبرمجيات التنقل المؤسسية.

وتقيمها مقابل الوضع ، Tracks A, B, C, D) يقوم هذا التقرير المخرجات الشاملة المطلوبة، حيث يقوم بتفكيك المسارات المقترحة "Shim" وتجمّع خارطة طريق تحرّم القيود غير القابلة للتفاوض للمشروع: معماريات DW\_workspace الحالي لمساحة العمل مباشرة في طبقة التطبيق، والحفاظ على مجموعة الاختبارات الحالية SDK الصارمة، منع استيراد حزم

### تعريف المعيار: "Uber" طموح الوصول لمستوى 1.2

يعني الالتزام بمجموعة محددة من المتطلبات غير الوظيفية التي تتجاوز مجرد تحسين<sup>3</sup> "Uber" إن الطموح للوصول إلى مستوى يعني هذا، Delivery Ways، واجهة المستخدم. في سياق مشروع

- القدرة على التعامل مع قطاعات خدمات متعددة (نقل الركاب، الطعام، Rungtusanatham et al.<sup>4</sup>) المرونة التشغيلية.
- الذين يجادلون،<sup>4</sup> دون هشاشة معمارية. يعكس هذا النتائج الأكاديمية التي توصل إليها بأن البساطة المنضبطة في العمليات الأساسية هي شرط أساسي للمرونة التشغيلية.
- يجب أن تدير المنصة الحالة: **(Event-Driven Real-Time Consistency)** الاتساق الآني المعتمد على الأحداث يقاس بالملاي ثانية (Latency) (موقع السائق، حالة الطلب) عبر عملاء موزعين بوقت استجابة دون المخاطرة بحدوث تراجع في "Ride" القدرة على نشر تحديثات لقطاع.
- دون المخاطرة بحدوث تراجع في "Ride" القدرة على نشر تحديثات المنتجات<sup>5</sup>، حيث تتفاعل الوحدات المتميزة من خلال واجهات "Parcels". يرتبط هذا بمفهوم الوحدات في معماريات عائلات المنتجات<sup>5</sup>، حيث تتفاعل الوحدات المتميزة من خلال واجهات "Shims".

### منهجية التحليل: تثليث البيانات 1.3

الصارمة. يتم التثليث بين ثلاث نقاط بيانات متميزة لاستخلاص الرؤى (Gap Analysis) "يعتمد هذا التقرير منهجية "تحليل الفجوة

- إلى A مقسمة إلى مسارات من، Manus) الحال المثالية التي اقتراها وكيل الذكاء الاصطناعي: خطة D.
- الحالة الحالية للكود، والتي تتميز بتنفيذات جزئية وديون تقنية موروثة: DW\_workspace) الواقع.
- التي تعمل (القيود الصلبة) Clean Architecture, Banned Imports) المعيار (القيود الصلبة: القواعد الهندسية غير القابلة للتغيير كمعايير قبول لأي كود يتم دمجه في المستودع.

## الإطار النظري: المرونة الصارمة وهندسة الوكلاء المستقلين 2.

### كأساس معماري "Rigid Flexibility" نموذج 2.1

"Rigid Flexibility" إلى ما يُعرف في أدبيات إدارة العمليات بنموذج Delivery Ways تستند الفلسفة المعمارية التي يجب أن تحكم مشروع في العمليات الأساسية لا (Discipline) يقترح هذا النموذج، الذي أعيد فحصه تجريبياً، أن الفرض الصارم للانضباط<sup>4</sup>. يحد من المرونة، بل على العكس، هو الممكن الرئيسي لها.

في سياق هندسة البرمجيات لتطبيقات الـ Super-Apps:

- تتمثل في القواعد التي تمنع المطوريين (أو وكلاء الذكاء الاصطناعي) من استيراد مكتبات خارجية: (Rigidity) الصرامة هذا القيد يبدو معيناً للسرعة في البداية. App Layer) مباشرة داخل طبقة التطبيق.

- أو (Google Maps إلى Mapbox) تظهر عندما تقرر الشركة استبدال مزود الخرائط (مثلاً من: **Flexibility**) المرونة يمكن إجراء هذا التغيير الجذري دون إعادة كتابة ، "Shims" تغيير مزود الدفع. بفضل الصرامة الأولية في استخدام طبقة الـ Business Logic) سطر واحد في منطق العمل.

أي القدرة - (**Mix Flexibility**) ومرنة المزيج (Volume Flexibility) تشير الأبحاث إلى أن محاولة تحقيق المرنة في الحجم على تقديم خدمات متعددة مثل الركاب والطرواد - تتطلب استراتيجيات تصنيع (أو برمجة) تفصل بين المكونات الثابتة والمكونات يجب أن تُقيّم بناءً على مدى التزامها بهذا الفصل، وليس فقط بناءً على سرعة الإنجاز Manus المتغيرة<sup>6</sup> وبالتالي، فإن خطة

## في التنفيذ (Manus) دور وكيل الذكاء الاصطناعي 2.2

يتجاوز مجرد تقديم الإجابات النصية إلى تنفيذ المهام المعقدة (Action Engine) وفقاً للمواد البحثية<sup>1</sup>، محركاً للفعل، Manus يُعد إلى تحديات محددة<sup>8</sup> خاصة بـ (Context Engineering) "وأتمته سير العمل. ومع ذلك، تشير أدبيات "هندسة السياق

- على سياق متراكم. إذا لم يتم تعديله بـ "القيود الصلبة" بشكل متكرر ومستقر Manus نافذة السياق والذاكرة؛ يعتمد فإنه قد "ينسى" القواعد المعمارية لصالح الحلول الأسرع والأكثر شيوعاً في بيانات تدريبيه (والتي غالباً ما تكون أ��واً)، Prefix Spaghetti Code).
- إلى المخرجات (Input Tokens) نسبة المدخلات إلى المخرجات: في الوكلاء المستقلين، تكون نسبة الرموز المميزة للمدخلات عالية جدًا (100:1). هذا يعني أن تكلفة "تنكير" الوكيل بالقواعد المعمارية عالية، ولكنها ضرورية لتجنب الديون التقنية ليس كوصفة نهائية، بل كـ "مسودة أولية" تتطلب مراجعة بشرية صارمة لضمان أنها لم Manus الاستنتاج: يجب معاملة خطة "تتجاوز حدود "الصرامة" المطلوبة لتحقيق "المرنة

## البنية التحتية ونظام التصميم: (Track A) تحليل المسار أ.

### أكثـر من مجرد واجهة: (App Shell) قشرة التطبيق 3.1

يعتبر هذا المسار هو Design System & App Shell Implementation على (A) يركز المسار، Manus في خطة لا تعمل "قشرة التطبيق" مجرد حاوية للتنقل، بل هي "نظام التشغيل، Super-Apps، الأساس الحرج للمشروع بأكمله. في تطبيقات الـ (المصغر) الذي يدير التطبيقات الفرعية Verticals).

#### تحليل الفجوة بين الخطة والواقع:

- تشير البيانات إلى أن نظام التصميم "مفعل جزئياً". هذا يعني وجود حالة خطيرة تعرف بـ **DW\_workspace**: الواقع في (Hard-coded)، حيث توجد مكونات قديمة تعتمد على قيم لونية ومسافات ثابتة، ("انحراف التصميم Design Drift)، ومكونات جديدة تحاول استخدام المتغيرات.
- (Production-Ready Implementation)." يهدف المسار إلى "تطبيق جاهز للإنتاج: Manus طموح

- يجب أن تكون معزولة تماماً عن منطق الأعمال، (Rides, Food) الرؤية المعمارية: لكي تنجح القشرة في دعم مسارات متعددة الخاص بتلك المسارات. يجب أن لا تعرف القشرة "كيف" يتم حجز رحلة، بل تعرف فقط "كيف" تعرض واجهة الرحلات.

## 3.2 معضلة الـ Shims وحظر الـ SDKs

"هذا نمط معماري يُعرف بـ "المنافذ والمحولات" (Ports and Adapters) أحد أهم القيد المذكور في التذكرة هو أو العمارة النظيفة (Clean Architecture)".

### 3.2.1 الضرورة النظرية والعملية

مباشرة في واجهات المستخدم. هذا يخلق افتراضًا وثيقاً package:firebase\_auth التقليدي، من الشائع استيراد Flutter في تطوير (Tight Coupling).

تغير مزود المصادقة، فإن التطبيق بأكمله يحتاج إلى إعادة كتابة Uber إذا قررت.

تقترح الخطة (ويجب أن تفرض) طبقة Shim Layer:

- تُعرف واجهات Dart تحتوي فقط على منطق (app/lib) طبقة التطبيق (Abstract Classes) لـ IMapProvider (مثل).

- فعليّة SDKs تنفذ هذه الواجهات باستخدام (packages/shims) طبقة المحولات.

مخاطر التنفيذ عبر الذكاء الاصطناعي:

(Leaky Abstractions) "في سعيه لسرعة الإكمال، قد يولد "تجريادات مسربة، Manus هو أن وكيل (A) الخط الرئيسي في المسار فإن التجريد GoogleMapController تحتوي على دالة تُرجع IMapProvider مثل: إذا كانت واجهة moveCamera(Waypoint) وهي وفاشل. يجب أن تكون الواجهة محابدة تماماً (مثل moveCamera(TargetLocation))."

## 3.3 نظام التصميم: الرموز مقابل القيم الثابتة

على نظام (A) في التناسق البصري، يجب أن يعتمد المسار Uber للوصول إلى مستوى "Design Tokens".

- يجب أن (C و B) في المسارات اللاحقة Manus التوصية: يجب أن يتضمن تقرير المراجعة بنداً يفترض أن أي كود يولد بل يجب استخدام Colors.blue يستهلك هذه الرمز. لا يُقبل context.theme.primaryBrandColor.

## 4. (Ride Vertical) تحليل المسار بـ (Track B) عمود التنقل والرحلات

## التنقل "MVP" التعقّيد الكامن في 4.1

الحد الأدنى للمنتج القابل للتطبيق (MVP) في عالم تطبيقات التنقل، مصطلح Ride Vertical MVP". تنفيذ (B) يتناول المسار "مضلل. التطبيق الذي يفشل في تحديد الموقع بدقة أو حساب السعر لا يعتبر "قابلاً للتطبيق".

## 4.2 آلة الحالة المحدودة (Finite State Machine - FSM)

جوهر تطبيق التنقل هو "آلة حالة" معقدة تدير دورة حياة الرحلة:

Draft -> Quoting -> Requesting -> Finding Driver -> Driver Accepted -> Driver Arrived -> In Progress -> Payment -> Complete.

تحليل الفجوة:

- الوضع الحالي: تشير المدخلات إلى وجود "واجهة مستخدم بسيطة + هيكل تتبع". هذا يعني أن المكونات المرئية موجودة، لكن هش أو مفقود (State Machine) ("المنطق" الذي يربطها بآلية الحالة).
- حيث الشبكة متصلة دائمًا والمسائق يقبل الطلب (Happy Path) "غالباً" المسار السعيد Manus يفترض: افتراضات فوراً.
- وليس في واجهات المستخدم (Domain Layer) التحدي الهندسي: يجب أن يكون منطق إدارة الحالة موجوداً في طبقة المجال (UI Widgets)، وهذا يسمح باختبار الحالات المعقّدة (مثل انقطاع الشبكة أثناء الرحلة) عبر اختبارات الوحدة (Unit Tests)، مما يساهم في الحفاظ على هدف "158 اختباراً".

## 4.3 تحدي الخرائط الجغرافية وتكامل الـ Shims

(A) التي تم تأسيسها في المسار Shims يعتبر التعامل مع الخرائط هو الاختبار الحقيقي لمعمارية الـ.

- تنفيذ خريطة تفاعلية عبر طبقة عزل هو أمر صعب تقنياً. الخرائط بطبيعتها أدوات مرئية وتفاعلية: مشكلة Map Shim.
- للأحداث (مثل النقر على الخريطة) ومصارف (Streams) وواجهة تعتمد على التدفقات Shim الحل المقترن: يجب أن يقوم الـ هو الوحيد الذي يضمن فصلاً حقيقياً (Reactive Pattern) للأوامر ( مثل تحريك الكاميرا). هذا النمط التفاعلي (Sinks).

## 4.4 محرّكات التسعير (Pricing Engines)

من الخادم (Quote) "يتم طلب "عرض سعر (Client-side). لا تحسب الأسعار على العميل Uber تطبيقات مستوى

- تحاكي منطق "Stub Service" إنشاء (B) جاهزاً، يجب أن يتضمن المسار (Backend) الفجوة: إذا لم يكن الخادم الخلفي التسuir وتأخير الشبكة. هذا ضروري لاختبار تجربة المستخدم في حالات "ارتفاع الطلب" أو "فشل الاتصال".
- 

## 5. التوسيع اللوجستي (Track C): تحليل المسار ج. (Parcels & Food)

### 5.1 الوحدات النمطية وفصل المجالات (Domain Separation)

من المغربي إعادة استخدام (Code Contamination). خدمات "الطرود والطعام". الخطأ هنا هو "تلوث الكود (C)" يقدم المسار لكنهما مختلفان جوهرياً. الرحلة لها "ركاب"، والطرود لها "أبعاد" و"تعليمات" لـ `ParcelDelivery` لتتمثل `RideRequest` كائن "استلام".

رؤيا نظرية: مرونة المزدوج (Mix Flexibility)

بالعودة إلى المراجع 5، تعني مرونة المزدوج قدرة النظام على التبديل بين أنواع مختلفة من المنتجات بأقل تكلفة.

- واحدة ملائمة بالحقول الفارغة `Order` (`Null Fields`).
  - قد تشتراك في (`RideOrder`, `FoodOrder`, `ParcelOrder`) العماره السينية: فئة **Manus**:
    - لـ `BaseOrder` سمات خفيفة لسجل الطلبات، ولكنها تتصل منفصلة في تدفق العمل النشط (`BaseOrder`)

### 5.2 لوحة تحكم التطبيق الشامل (Super App Dashboard)

الشاشة الرئيسية تحتاج إلى توجيه المستخدمين بنكاء إلى هذه المسارات.

- " الواقع: "شاشات تجريبية" (`Demo Screens`).
    - تنفيذ المنطق **Manus**: خطوة هذا يتطلب.
  - (C) يجب أن تكون متاحة في المسار (B) الفجوة: يجب معالجة "الموارد المشتركة". طريقة الدفع المختارة في المسار تقع تحت التطبيقات العمودية (`Horizontal Module`) أن تكون ميزة "الدفع" وحدة مشتركة.
- 

## 6. بوابات العبور والهوية الرقمية (Track D): تحليل المسار د. (Auth & Onboarding)

## 6.1 (Identity Shim) إدارة الهوية كطبقة عزل

أو Firebase Auth أو AWS Cognito. هي أهم طبقة عزل. يجب ألا يعرف التطبيق ما إذا كان يستخدم (Authentication) المصادقة (DSR).

- **الخطر:** الوضع الحالي يشير إلى أنها "موجودة جزئياً في الكود/الدراسات". هذا هو أخطر وضع، حيث أن كود المصادقة نصف المكتوب هو ثغرة أمنية لأنه لا يمكن ، (A) أو حتى بالتوافق مع المسار (C) ليكون قبل المسار (D) استراتيجية التنفيذ: يقترح التقرير نقل تنفيذ المسار (B) حجز رحلة (User ID).

## 6.2 (DSR) طلب خدمة السائق

(تتضمن هذه الميزة تدفقات معقدة لرفع المستندات (رخصة القيادة، صور السيارة).

- **التعامل مع الوسائط:** يتطلب رفع الصور التعامل مع الأذونات (الكاميرا/المعصر في permission\_handler أي استدعاء مباشر لمكتبة PermissionShim.
- **فحص القيود:** يجب التعامل مع الأذونات عبر واجهة المستخدم بعد خرقاً للمعمارية حيث تطلب المعلومات ، ("التمييز التدريجي" Uber معيار تجربة المستخدم؛ تستخدم تطبيقات للتأكد من اتباع هذا النمط Manus على مراحل. يجب مراجعة خطة

## 7. Comprehensive Gap Analysis (تحليل الفجوات الشامل)

والواقع، وتُستخدم لتوليد ملف Manus تهدف الجداول التالية إلى تحديد الفجوات بين خطة docs/analysis/manus\_tracks\_gap\_analysis.md.

### 7.1 جدول مقارنة المسار A (Design & Shell)

المعيار	افتراضات Manus	واقع DW_workspace	الفجوة (Gap)	الإجراء التصحيحي

<b>نظام التصميم</b>	نظام مكتمل وجاهز للاستخدام	مفعلاً جزئياً، خليط من القديم والجديد	عدم وجود استراتيجية لترحيل الشاشات القديمة	إضافة مهمة "Migration Sprint" لتحويل الشاشات القديمة
<b>الهيكلية (Navigation)</b>	استخدام go_router أو مشابه مع Shell Route	كود ملاحقة قديم (Legacy Navigator)	تعرض في إدارة الـ Deep Links وحالة القشرة	إعادة كتابة RouterDelegate بالكامل في Track A
<b>الاستقلالية</b>	القشرة لا تعتمد على التفاصيل	القشرة قد تكون حالياً Auth مقترنة بـ	القشرة تحتاج لمعرفة "حالة الدخول" لعرض القوائم	فصل AuthStatusStream كواجهة وحيدة للقشرة

## 7.2 جدول مقارنة المسار (Ride Vertical)

المعيار	افتراضات Manus	واقع DW_workspace	الفجوة (Gap)	الإجراء التصحيحي
<b>إدارة الحالة (State)</b>	تدفق سعيد (Happy Path)	واجهة مستخدم بسيطة دون منطق ربط	غياب التعامل مع حالات الحافة (Edge Cases) وانقطاع الشبكة	فرض اختبارات "Chaos Testing" على الـ FSM
<b>الخرائط (Maps)</b>	تكامل جاهز (Plug & Play)	لا يوجد Map Shim حقيقي	الخرائط الحالية قد تكون Hard-coded	بناء MapInterface UI قبل أي كود
<b>التسعير (Pricing)</b>	واجهة برمجة تطبيقات API (API) جاهزة	غير معروف / غير موجود	التطبيق سيفشل عند محاولة جلب السعر	بناء MockPricingService داخل Dart لمحاكاة الخادم

## 8. توصيات التنفيذ وخارطة الطريق (Execution Roadmap)

والحاجة إلى ضوابط بشرية، نوصي بتعديل استراتيجية التنفيذ Manus بناءً على التحليل أعلاه، وبالنظر إلى الطبيعة "المستقلة" لوكيل

### 8.1 ترتيب التنفيذ المقترن (Re-sequencing)

بدون "مستخدم (B)" الذي تقتربه الخطة يحمل مخاطر عالية. لا يمكن اختبار "جزء الرحلة (D -> C -> B -> A)" الترتيب الخطى "مسجل (D)".

الترتيب الموصى به:

الأساس): بناء القشرة ونظام التصميم. هذا شرط مسبق لا يمكن تجاوزه) **Track A**.

مستقبلي API حقيقى للتعامل مع أي AuthToken الهوية - مقدم): تنفيذ المصادقة الأساسية. تحتاج إلى) **Track D**.

تنفيذ القيمة الأساسية للمشروع - جوهر العمل) **Track B - Rides**:

يمكن تأجيله أو تنفيذه بالتزامن إذا توفرت الموارد - التوسيع) **Track C - Logistics**:

### 8.2 إدارة المخاطر والأسئلة المفتوحة

للتتنفيذ، يجب حسم النقاط التالية (لتضمينها في Manus قبل إطلاق manus\_execution\_recommendations.md):

- **مخاطر "الديون الخفية" للذكاء الاصطناعي**:

للاستيراد aliasing عن طريق عمل "No SDKs" بتوليد كود يعمل ظاهريًا ولكنه يخرق قاعدة Manus الخطر: قد يقوم ○ أو وضعه في ملفات غير مرئية.

عند اكتشاف أي (Build Failure) ليرفض البناء analysis\_options.yaml في ملف custom\_lint في حل: تكوين ○ استيراد محظوظ، مما يجر الوكيل على التصحيح الذاتي.

- **سؤال حول جاهزية Backend:**

بكتابه Manus جاهزة؟ إذا كانت الإجابة "لا"، يجب تكليف WebSocket لخدمة RideMatching هل واجهات الـ ○ كجزء من المسار Dart بلغة (Mock Server) ("خادم محاكاة B).

- **(Feature Flags): استراتيجية الميزات**:

التكوين (Track A) "في الإطلاق الأول؟ إذا نعم، يجب أن تدعم "القشرة (Flag) خلف ميزة "Parcels" هل نريد إخفاء ○ لإخفاء أيقونة الطرود (Remote Config) عن بعد.

### 8.3 التوافق مع المعايير الداخلية

"Clean B Workspace": تضمن هذه الخطة المعدلة الالتزام بمعايير

- **فقط:** يتم عزل كل مكتبة خارجية Shims التكامل عبر الـ PR وليس السقف. أي (Floor) الحفاظ على الاختبارات: يتم اعتبار رقم "158 اختباراً" هو الأرضية.
  - يقلل النسبة يتم رفضه PR لإدارة الحزم المتعددة، مما يفرض الفصل المادي بين الكود melos نظافة المستودع: يتم استخدام
- 

## 9. الخاتمة.

باستخدام قدرات الذكاء الاصطناعي المتقدمة "Uber" فرصة فريدة للفوز نحو مستوى "Delivery Ways" يواجه مشروع (Manus). ومع ذلك، يُظهر هذا التحليل أن الاعتماد الأعمى على الخطة المقترحة دون حوكمة معمارية سيؤدي إلى منتج "سريع". الانكسار" بدلاً من "من

وفصل المجالات - هو الطريق الوحيد Shims من خلال فرض قيود صارمة على الـ "Rigid Flexibility" إن تطبيق نموذج (D) لا تتحول إلى ديون تقنية طويلة الأمد. من خلال إعادة ترتيب المسارات لتقدير المصادقة Manus لضمان أن السرعة التي يوفرها وتوسيع استراتيجية محاكاة الخدمات الخلفية، يمكن للفريق تحويل هذه الخطة النظرية إلى واقع برمجي قوي وقابل للتلوّع.

---

## (Simulated Output Files) الملاحقات: محتوى ملفات المخرجات

جاهز للحفظ في Markdown فيما يلي المحتوى التفصيلي للملفات الثلاثة المطلوبة، مصاغاً بتنسيق docs/analysis/.

الملف الأول: [docs/analysis/manus\\_plan\\_overview.md](#)

# مشروع Manus نظرة عامة على خطة تطوير Delivery Ways

## المُلخص التنفيذي

من نموذج أولى إلى منصة تنقل متعددة الخدمات جاهزة Manus DW\_workspace تهدف خطة مع التركيز على إنشاء قشرة تطبيق ، (A, B, C, D) تقوم الخطة بتفكيك العمل إلى أربعة مسارات رئيسية ("مستوى Uber" للإنتاج (App Shell) قبل التوسيع في التطبيقات العمودية المعقّدة. الخطة طموحة وتقترض جاهزية عالية للخدمات الخلفية (Backend). وتتطلب التزاماً صارماً بالقيود المعمارية لضمان الجودة ،

## هيكلية المسارات (Tracks Breakdown)

(Track) المسار	العنوان في الخطة	الهدف الاستراتيجي	أهم المخرجات (Deliverables)
A	<b>Design System &amp; App Shell</b>	تأسيس البنية التحتية المرئية والملاحة.	<ul style="list-style-type: none"><li>قشرة تطبيق جاهزة للإنتاج ((ملاحة، توجيه))</li><li>نظام تصميم قائم على (Tokens) الرموز</li><li>محرك الثيمات (Theme Engine)</li></ul>
B	<b>Ride Vertical MVP</b>	تنفيذ الوظائف الأساسية لخدمة النقل.	<ul style="list-style-type: none"><li>آلية حالة الرحلة (FSM)</li><li>تكامل خرائط Uber Shim</li><li>واجهة التسuirir والجزء</li></ul>
C	<b>Parcels &amp; Food MVP</b>	توسيع المنصة لتشمل الخدمات اللوجستية.	<ul style="list-style-type: none"><li>معالجات نماذج الطرود (Wizards)</li><li>واجهة كتالوج الطعام</li><li>لوحة تحكم متعددة الخدمات</li></ul>
D	<b>Onboarding, Auth,</b>	إدارة الهوية وجانب العرض	<ul style="list-style-type: none"><li>Shim آمن (OIDC/OAuth)</li></ul>

	<b>DSR</b>	((السائقين)).	<ul style="list-style-type: none"> <li>تدفق طلب خدمة السائق (DSR)</li> <li>إدارة الملف الشخصي</li> </ul>
--	------------	---------------	--

## القيود الصلبة (Hard Constraints)

تم تحديد القيود التالية كبوابات لا يمكن تجاوزها لضمان نجاح المشروع:

- (غير قابلة للتفاوض) Shims عمارة الـ:**
  - داخل مجلد (Firebase, Google Maps, Stripe) القيد: يمنع منعًا باتًّا استيراد حزم البنية التحتية الخارجية (مثل app/lib).
  - معرفة في طبقة المجال (Domain) المتطلب: يجب أن تمر جميع التفاعلات مع الطرف الثالث عبر واجهات مجردة (Domain) ومطبقة في packages/shims.
- خط الأساس للاختبارات:**
  - القيد: يجب أن تتجه مجموعة الاختبارات الحالية (158 اختبارًا) في جميع الأوقات.
  - يجب أن تحافظ على نسبة التغطية أو تزيدها (Tracks B/C) المتطلب: الميزات الجديدة.
- الصارمة قواعد Linter:**
  - عند وجود استيرادات محظورة (Errors) لإطلاق أخطاء analysis\_options.yaml القيد: يجب تكوين.
- (Build Health): صحة البناء**
  - المستقرة دون الاعتماد على مسارات محلية مطلقة Flutter القيد: يجب أن يتم بناء المشروع على قنوات.

## مع الواقع الحالي v3.2.1 مقارنة حالة المشروع

- منفذة جزئياً "RBAC" إلى أن "الإشعارات" و"التحكم في الوصول v3.2.1 نطاق التقرير: يشير تقرير.
- التعارض:** مساحة العمل الحالية تحتوي على هذه الميزات في حالة "مفاهيمية" ولكنها غير مدمجة بالكامل في قشرة التطبيق الجديدة A.
- الرحلات) بشكل كبير على الإشعارات لتحديثات الحالة (مثل "وصل السائق"). إذا لم تكن الإشعارات) B التبعية: يعتمد المسار B سيفشل المسار Shim، مغلفة بطبقة.

الملف الثاني: [docs/analysis/manus\\_tracks\\_gap\\_analysis.md](docs/analysis/manus_tracks_gap_analysis.md)

# تحليل الفجوات التفصيلي لمسارات Manus

الواقع الفعلي لمستودع Manus يقارن هذا المستند بين الافتراضات الواردة في خطة DW\_workspace.

## Track A: Design System & App Shell Implementation

- **TL;DR:** إعادة بناء MaterialApp "جزء من Super App" الذي يدعم التنقل لـ "MaterialApp". تصميم قائم على الرموز.
- **افتراضات Manus (What Manus Assumes):** بسهولة دون كسر جميع الشاشات الحالية (Root Widget) أنه يمكننا إعادة هيكلة عنصر واجهة المستخدم الجذري.
  1. بسهولة دون كسر جميع الشاشات الحالية (Root Widget) أنه يمكننا إعادة هيكلة عنصر واجهة المستخدم الجذري.
  2. وليس نموذج حوكمة للرموز، Widgets هو مجرد مجموعة من الـ Tokens.
- **الواقع الحالي (DW\_workspace Reality):** نظام تصميم جزئي حيث تستخدم بعض العناصر الرموز والبعض الآخر يستخدم ألوانًا ثابتة.
  1. نظام تصميم جزئي حيث تستخدم بعض العناصر الرموز والبعض الآخر يستخدم ألوانًا ثابتة.
  2. أو بنية القشرة المقترنة go\_router مقدس تنقل قيم قد لا يكون متوافقًا مع.
- **(Gaps & Contradictions):** لأكثر من 50 شاشة موجودة، إذا قام (Migration Strategy) "الفجوة" بـ "الفجوة" لا تذكر الخطة صراحة "استراتيجية الترحيل" (Franken-UI).
  - واجهة مستخدمة مكونات جديدة فقط، فستنتهي بواجهة مستخدمة هيكلة A المسار.
  - والوضع الليلي، لكن العينات المقدمة (Accessibility) التناقض: مصطلح "جاهز للإنصال" يعني دعم إمكانية الوصول.
- **عمل ثقيل vs مكاسب سريعة:**
  - ورموز الألوان فورًا ThemeData مكسب سريع: تعريف.
  - عمل ثقيل: إعادة هيكلة منطق التنقل لفصل القشرة تمامًا عن التطبيقات العمودية.

## Track B: Ride Vertical MVP Implementation

- **منطق العمل الأساسي.** حجز رحلة، تتبع السائق، وإدارة جلسة الرحلة.
- **افتراضات Manus:** والتحكم بالكاميرا (Markers) والخطوط (Polylines) يعمل ويدعم العلامات "Map Shim" وجود.
  1. "Map Shim" وجود.
  2. يدفع تحديثات "حالة الرحلة" عبر WebSocket/Stream.
- **الواقع الحالي:**
  - شاشات واجهة مستخدم بسيطة لعملية "الحجز".
    1. "Mock" من المحتمل أن يكون ثابتاً أو يعتمد على بيانات وهمية Tracking.
    2. "هيكل" تتبع.

- **الحالات الافتراضية**: مباشرة في الكود القديم Google Maps Widget بمستوى الإنتاج (غالباً ما يتم استخدام Map Shim لا يوجد).
- **الفجوات والتناقضات**:
  - يحتاج التطبيق للتعامل مع أحداث "انتهاء صلاحية السعر" أو "زيادة". (الفجوة: منطق "محرك التسعير Uber". الخطة الحالية تشير لتدفق بسيط "عرض السعر -> حجز"، وهو ساذج جداً لمستوى الأسعار.)
  - إذا تم إغلاق التطبيق أثناء الرحلة، فإن منطق "إعادة التمية" (State Loss) (Rehydration) الخطير: فقدان الحالة لحالة الرحلة من الخادم عند بدء التشغيل مجدد و غالباً ما يتم تجاهله في خطط MVP.
- **عمل ثقيل vs مكاسب سريعة**:
  - باستخدام نظام التصميم الجديد (Booking Sheet) "مكاسب سريعة: تحسين واجهة" ورقة الحجز.
  - الذي يتعامل مع أكثر من 15 حالة لدورة حياة الرحلة بقوة RideBloc و RideRepository عمل ثقيل: بناء.

## Track C: Parcels & Food MVP Implementation

- **TL;DR**: إضافة القدرات اللوجستية والتجارية.
- **افتراضات Manus**:
  - يمكن للطرود والطعام مشاركة نفس هيكل "الطلب" (افتراض خطير).
    1. مطابق للرحلات (Checkout) تدفق الدفع.
    2. للطرود (Demo) شاشات تجريبية.
- **الواقع الحالي**:
  - للطعام. الرحلات ليس لديها سلة؛ الطعام لديه. هذا يتطلب آلية تخزين (Basket Management) "الفجوة: إدارة السلة".
  - محلية جديدة عامة لـ "السلة".
- **الفجوات والتناقضات**:
  - لعنوان التسليم في المسار B من المسار (Location Picker) "مكاسب سريعة: إعادة استخدام" منتقى الموقع.

## Track D: Onboarding, Auth, and DSR

- **TL;DR**: الهوية، الأمان، واكتساب السائقين.
- **افتراضات Manus**:
  - المصادقة هي مجرد "شاشة تسجيل دخول".
    1. "هو مجرد نموذج DSR".
    2. "الصادقة هي مجرد شاشة تسجيل دخول".
- **الواقع الحالي**:
  - منطق مصادقة منفذ جزئياً (غالباً بدون معالجة تجديد التوكن الكاملة).
- **الفجوات والتناقضات**:
  - إذا فشل استدعاء API في تطبيق بمستوى Uber ، يجب أن يقوم التطبيق بتجديد التوكن (Token Refresh Rotation). (الفجوة: تدوير تجديد التوكن)
    - هذا ينتمي (Interceptor) "برمز 401" ، يجب أن يقوم التطبيق بتجديد التوكن بصمت وإعادة المحاولة. منطق "المعترض" ولكن يؤثر على المصادقة HTTP Shim لطبقة.

- الرحلات) يحتاج لمستخدم موثق لإنشاء حجز) B لكن المسار ،(D) التناقض: تضع الخطة هذا المسار أخيراً ○

الملف الثالث: [docs/analysis/manus\\_execution\\_recommendations.md](#)

## توصيات التنفيذ وخارطة الطريق

### 1. مقتراح ترتيب التنفيذ (Execution Order Proposal)

التسلسل الموصى به: Track A → Track D → Track B → Track C

- الأساس): لا يمكننا بناء ميزات بدون الفشرة ونظام التصميم، إنه يفرض القواعد المعمارية (Track A).
- صالح (Auth Token) وتوكن مصادقة (User Entity) الهوية/المصادقة): تم تقديمها. تحتاج إلى كيان مستخدم (Track D) بمستخدمين وهما يخلق ديواناً تقنية B لأي تطبيق عمودي. بناء المسار Backend للتفاعل مع الـ
- (Sockets) ومقابس الوقت الحقيقي Map Shim الرحلات): القيمة الأعلى. يتحقق من صحة Track B (MVP
- الطرود/الطعام): توسيع في التعقيد. يمكن العمل عليه بالتوازي بمجرد ترسيخ الأنماط في المسار B.

### 2. المخاطر والأسئلة المفتوحة (Risks & Open Questions)

المخاطر الاستراتيجية:

- أغلفة للخرائط، المدفوعات، المصادقة) يستغرق وقتاً أطول بمرتين من التكامل) Shims تطوير الـ "Shim": عنق زجاجة الـ المباشر.
- الميزات Shims قبل Sprints ببنية تحتية" محددة فقط لتطوير الـ "Sprints" التخفيف: تخصيص ○ إذا كانت الجلسة طويلة جدًا "Banned Imports" قد يفقد وكيل الذكاء الاصطناعي سياق قاعدة: Manus حدود سياق.
- يخرج القاعدة، مما يجبر الوكيل على تصحيح نفسه PR لرفض أي Linter CI/CD) صارم التخفيف: استخدام ○

أسئلة مفتوحة للجسم:

- استراتيجية الطرود والطعام: هل نريد إتاحتها للجمهور فوراً، أم يجب إخفاوها خلف Feature Flag (enable\_logistics\_vertical)؟

- جاهزة؟ إذا لم تكن كذلك، يجب بناء RideMatching Socket API هل واجهة: *Backend*
2. لفاف الحظر عن تطوير المسار MockRideService في Dart B.
  3. الآن لدعم التبديل بينهما لاحقاً Shim ؟ يجب تصميم واجهة Google Maps مقابل Mapbox مزود الخرائط: هل قررنا مؤتمته بالكامل أم تتطلب لوحة إدارة (Track D) هل عملية انضمام السائقين: *DSR* من Admin Panel؟ إذا كانت
  4. بيدوية، تحتاج لمعرفة من سيتحقق من السائقين أثناء الاختبار.

## 3. التوافق مع المعايير الداخلية (Alignment with Internal Standards)

"Clean B Workspace": ترفض خطة التنفيذ هذه بصرامة معايير

- اعتماد infrastructure: ابدأ على app لا يعتمد UI و domain على app قاعدة التبعية: يعتمد اختبارات وحدة. خط الأساس "158 اختباراً" هو A-D يتم إنشاؤه للمسارات PR الحفاظ على الاختبارات: يجب أن يتضمن كل الحد الأدنى مما يضمن الفصل ،
- **Monorepo:** سيتتم استخدام melos لإدارة الحزم المنفصلة (app, shims, features/\*) صحة الـ المادي للوحدات على نظام الملفات.

### Works cited

1. Manus AI: General AI Agent, accessed November 27, 2025, <https://manus.so/>
2. #1 Manus AI - The World's First General-Purpose AI Assistant Launching in 2025, accessed November 27, 2025, <https://www.manusai.io/>
3. How to Build a Taxi App like Uber in 2024: Features and Cost - Django Stars, accessed November 27, 2025, <https://djangostars.com/blog/how-much-does-it-cost-to-build-a-taxi-app-like-uber/>
4. Effects of simplicity and discipline on operational flexibility: An empirical reexamination of the rigid flexibility model | Request PDF - ResearchGate, accessed November 27, 2025, [https://www.researchgate.net/publication/247174124\\_Effects\\_of\\_simplicity\\_and\\_discipline\\_on\\_operational\\_flexibility\\_An\\_empirical\\_reexamination\\_of\\_the\\_rigid\\_flexibility\\_model](https://www.researchgate.net/publication/247174124_Effects_of_simplicity_and_discipline_on_operational_flexibility_An_empirical_reexamination_of_the_rigid_flexibility_model)
5. The Concepts and Determinants of Manufacturing Flexibility | Request PDF - ResearchGate, accessed November 27, 2025, [https://www.researchgate.net/publication/362942471\\_The\\_Concepts\\_and\\_Determinants\\_of\\_Manufacturing\\_Flexibility](https://www.researchgate.net/publication/362942471_The_Concepts_and_Determinants_of_Manufacturing_Flexibility)
6. Considerations on a contemporary Flexibility Approach | Request PDF - ResearchGate, accessed November 27, 2025, [https://www.researchgate.net/publication/234092839\\_Considerations\\_on\\_a\\_contemporary\\_Flexibility\\_Approach](https://www.researchgate.net/publication/234092839_Considerations_on_a_contemporary_Flexibility_Approach)
7. Manus: Hands On AI, accessed November 27, 2025, <https://manus.im/>

8. Context Engineering for AI Agents: Lessons from Building Manus, accessed November 27, 2025,  
<https://manus.im/blog/Context-Engineering-for-AI-Agents-Lessons-from-Building-Manus>