### **خريطة عمل وخطة تطبيق "Plasticology" (النسخة النهائية الكاملة)**

هذا المستند هو المرجع الفني الشامل للتطبيق، يوضح هيكل البيانات، بنية الكود، وتدفق المعلومات بين الواجهة الأمامية والخادم.

#### **1. الهدف الأساسي للتطبيق**

منصة تعليمية تفاعلية متكاملة لجراحي التجميل، تهدف إلى توفير كل ما يحتاجه الطبيب للتحضير للامتحانات والممارسة العملية، من خلال دمج المحاضرات العلمية مع بنوك الأسئلة المتقدمة، الحالات الإكلينيكية، وأدوات ذكية لتنظيم الدراسة وتتبع الأداء.

#### **2. هيكل قاعدة البيانات (Backend - Google Sheets)**

يعتمد التطبيق على ملف Google Sheets واحد يحتوي على عدة شيتات (Sheets)، كل منها يمثل جدول بيانات لغرض معين. يجب أن تكون أسماء الأعمدة مطابقة تمامًا لما هو مكتوب أدناه.

* **Sheet: Users**
  + **الوظيفة:** إدارة بيانات المستخدمين الأساسية، صلاحياتهم، وتفاصيل اشتراكاتهم.
  + **الأعمدة:** UniqueID, Name, Username, Password, Role, UserIMG, AccessGranted, AdditionTimeStamp, CourseChosen, SubscriptionDuration, SubscriptionEndDate, StudyType, E-Mail, MobileNumber, Country, ExamDate, AdminNotes, Exam Results, Payment, PaymentMethod.
* **Sheet: Role**
  + **الوظيفة:** تحديد صلاحيات الوصول لكل دور (Role) في التطبيق. إذا كانت القيمة TRUE (أحرف كبيرة)، يتم تفعيل الميزة.
  + **الأعمدة:** Unique\_ID, Role, Lectures, MCQBank, LerningMode, OSCEBank, LeadersBoard, Radio, Library, StudyPlanner.
* **Sheet: Users\_Card**
  + **الوظيفة:** تخزين البيانات الشخصية للمستخدم التي يمكنه تعديلها، وحساب مجموع نقاطه تلقائيًا.
  + **الأعمدة:** UniqueID, Name, Nickname, ExamDate, Age, StudyType, QuizScore, User\_Img.
* **Sheet: MCQ Bank**
  + **الوظيفة:** بنك الأسئلة الرئيسي للمشتركين.
  + **الأعمدة:** UniqueID, Chapter, Question, Hint, CorrectAnswer, IncorrectAnswer1, IncorrectAnswer2, IncorrectAnswer3, CorrectRationale, IncorrectRationale1, IncorrectRationale2, IncorrectRationale3, Source, ImageURL, AudioURL.
* **Sheet: FreeTestMCQs**
  + **الوظيفة:** بنك أسئلة مخصص للاختبار المجاني للزوار.
  + **الأعمدة:** UniqueID, Chapter, Question, Hint, CorrectAnswer, IncorrectAnswer1, IncorrectAnswer2, IncorrectAnswer3, CorrectRationale, IncorrectRationale1, IncorrectRationale2, IncorrectRationale3, Source, ImageURL, AudioURL.
* **Sheet: Lectures Topics**
  + **الوظيفة:** تنظيم المحاضرات وفيديوهاتها وربطها بالفصول.
  + **الأعمدة:** UniqueID, Chapter, LectureName, LectureURL, Mock Name, Mock Link, ChapterIcon.
* **Sheet: OSCE\_Cases**
  + **الوظيفة:** يحتوي على تفاصيل الحالات الإكلينيكية وصورها.
  + **الأعمدة:** CaseID, Title, ImageURL, Hint, CaseDescription, AudioURL, Chapter, Source.
* **Sheet: OSCE\_Questions**
  + **الوظيفة:** يحتوي على الأسئلة المرتبطة بكل حالة عبر CaseID.
  + **الأعمدة:** QuestionID, CaseID, QuestionType, QuestionText, CorrectAnswer, IncorrectAnswer1, IncorrectAnswer2, IncorrectAnswer3, CorrectRationale, IncorrectRationale1, IncorrectRationale2, IncorrectRationale3, EssayModelAnswer, Source, ImageURL, AudioURL.
* **Sheet: Study\_Planner**
  + **الوظيفة:** تخزين خطط المذاكرة المتعددة الخاصة بكل مستخدم.
  + **الأعمدة:** User\_ID, Plan\_Name, Plan\_StartDate, Plan\_EndDate, Study\_Plan\_JSON, Plan\_ID, Plan\_TimeStamp, Plan\_Status.
* **Sheet: QuizLogs**
  + **الوظيفة:** تسجيل نتائج كل اختبار يقوم به المستخدم.
  + **الأعمدة:** Timestamp, UserID, UserName, EventType, QuizTitle, Score, TotalQuestions, Details.
* **Sheet: LectureLogs**
  + **الوظيفة:** تسجيل كل محاضرة يقوم المستخدم بمشاهدتها.
  + **الأعمدة:** Timestamp, UserID, UserName, EventType, LectureName.
* **Sheet: AnsweredQuestionsLog**
  + **الوظيفة:** تسجيل كل سؤال فريد يجيب عليه المستخدم (لمعرفة الأسئلة التي لم يتم حلها).
  + **الأعمدة:** UserID, QuestionID.
* **Sheet: UserIncorrectQuestions**
  + **الوظيفة:** سجل ذكي لتتبع أخطاء المستخدم في الأسئلة.
  + **الأعمدة:** UniqueID, UserID, QuestionID, UserAnswer, WrongCount, CorrectTimestamp, LastUpdate.
* **Sheet: UserQuizNotes & UserLectureNotes**
  + **الوظيفة:** تخزين الملاحظات النصية.
  + **أعمدة ملاحظات الاختبار:** UniqueID, Timestamp, UserID, QuizID, NoteText.
  + **أعمدة ملاحظات المحاضرات:** UniqueID, Timestamp, UserID, LectureID, NoteText.
* **Sheet: MCQBooks, Announcements, Messenger**
  + **الوظيفة:** شيتات مساعدة للمكتبة، الإعلانات، والرسائل.
  + **أعمدة الكتب:** UniqueID, Book, Description, Link, icon.
  + **أعمدة الإعلانات:** UniqueID, TimeStamp, UpdateMessage, IsActive.
  + **أعمدة الرسائل:** Timestamp, UserID, UserName, UserMessage, AdminReply.

#### **3. هيكل الواجهة الأمامية وترابط الملفات (Frontend Architecture)**

تم بناء التطبيق كتطبيق صفحة واحدة (SPA) باستخدام وحدات جافاسكريبت (ES Modules).

* **index.html**: الملف الأساسي الذي يحتوي على كل هياكل الشاشات والنوافذ المختلفة.
* **style.css**: ملف التنسيقات المخصصة (Custom CSS).
* **مجلد js/**: يحتوي على كل ملفات الجافاسكريبت، وترابطها كالتالي:  
  📁 js/  
  ├── 📄 main.js (المنظم الرئيسي: يستورد كل شيء ويربط الأزرار بالوظائف)  
  |  
  ├── 📄 api.js (جسر الاتصال: يُستورَد بواسطة كل ملفات features التي تحتاج بيانات من الخادم)  
  ├── 📄 dom.js (قاموس العناصر: يُستورَد بواسطة كل الملفات تقريبًا للوصول لعناصر الصفحة)  
  ├── 📄 state.js (الذاكرة: يُستورَد بواسطة كل الملفات تقريبًا للوصول إلى/تعديل بيانات التطبيق الحية)  
  ├── 📄 ui.js (مدير الواجهة: يُستورَد بواسطة main.js وملفات features لعرض الشاشات والنوافذ)  
  ├── 📄 utils.js (صندوق الأدوات: يُستورَد بواسطة الملفات التي تحتاج دوال مساعدة عامة)  
  |  
  └── 📁 features/ (مجلد الميزات المتخصصة)  
   ├── 📄 quiz.js (منطق اختبارات MCQ، يُستورَد بواسطة main.js و planner.js)  
   ├── 📄 osce.js (منطق اختبارات OSCE، يُستورَد بواسطة main.js)  
   ├── 📄 lectures.js (منطق المحاضرات، يُستورَد بواسطة main.js)  
   ├── 📄 planner.js (منطق منظم المذاكرة، يُستورَد بواسطة main.js)  
   ├── 📄 userProfile.js (منطق المستخدم، يُستورَد بواسطة main.js و ui.js)  
   ├── 📄 learningMode.js (منطق وضع التعلم، يُستورَد بواسطة main.js)  
   ├── 📄 notes.js (منطق الملاحظات، يُستورَد بواسطة main.js)  
   ├── 📄 activityLog.js (منطق سجل النشاطات، يُستورَد بواسطة main.js)  
   └── 📄 leaderboard.js (منطق لوحة الصدارة، يُستورَد بواسطة main.js)

#### **4. تدفق البيانات (Data Flow)**

1. **بدء التشغيل:** main.js يستدعي initializeApp التي تستدعي fetchContentData من api.js.
2. api.js يرسل طلب GET إلى doGet في Google Apps Script.
3. الخادم يجمع البيانات العامة (أسئلة، محاضرات، أدوار، أسئلة مجانية...) ويرسلها كـ JSON.
4. main.js يستقبل البيانات، يستخدم utils.js لتنسيقها، ويخزنها في appState داخل state.js. ثم يقوم بتعبئة الفلاتر الأولية باستخدام ui.js.
5. **تسجيل الدخول:** userProfile.js يرسل طلب POST ببيانات الدخول.
6. GS Code.txt في doPost يتحقق من المستخدم ويعيد بياناته.
7. بعد النجاح، يتم استدعاء fetchUserData من api.js لجلب كل البيانات الخاصة بالمستخدم (سجلاته، ملاحظاته، خططه، الأسئلة المجابة) وتخزينها في appState.
8. **التفاعل:** أي ضغطة زر يتم التقاطها في main.js الذي بدوره يستدعي الدالة المناسبة من ملف الـ feature المختص.

هذا المستند يمثل الخريطة الكاملة للتطبيق. في رسالتي التالية، سأقوم بشرح كل ميزة وزر بالتفصيل من وجهة نظر المستخدم.