



レジシステムとか飲食店の注文システムとか、だいたいどれくらい月額ライセンス料をとられるもの？

🗣️ 8秒間シンキングしました

だいたいの相場は、基本POSレジで月額0～1.5万円程度、注文システム（ハンディ・テーブルオーダー・モバイルオーダー）を加えると月額1～3万円前後が一般的です。店舗規模・機能・端末台数で大きく変わります。

1. 基本のレジシステム（POSレジ）の月額ライセンス料

クラウド型タブレットPOSが主流で、小規模店向けはかなり安く抑えられます。

- **無料～数千円**：Airレジ（基本無料）、Square（フリープラン0円）、STORESレジやfunfoの一部プランなど。基本的な会計・売上管理はこれで足りるケースが多い。
- **3,000～1万円前後**：MAIDO POS（2,980円～）、スマレジのプレミアム系（5,500～8,800円）、その他中規模向け。
- **1～1.5万円前後**：飲食店特化の高機能プラン（スマレジ フードビジネス 15,400円税込、Squareプラス13,000円など）。テーブル管理や分析機能が充実。

相場として、クラウド型POSは**月額0～1.5万円**が中心です。端末（iPadなど）や周辺機器（プリンター・ドロア）は別途初期費用がかかります（5～30万円程度）。

2. 飲食店の注文システム（オーダーエントリー）の月額

ハンディ端末、テーブル設置タブレット、QRモバイルオーダーなどが該当。POSと一体型かオプションかで料金が異なります。

- **ハンディ・キッチンモニター中心**：1～2万円前後（例：Airレジ オーダーでハンディ込み13,200～17,600円程度、台数追加で増額）。
- **テーブルオーダー／モバイルオーダー**：1～3万円前後が目安（例：テーブルオーダー単体で1万円～、モバイルオーダー併用で1.3万円～。サービスによっては5,000～5万円の幅あり）。





- **一体型・フル機能**：POS+オーダーで合計1.5～3万円程度になることが多い。

相場として「月額5,000～5万円程度に分布し、定額型と台数課金型がある」との指摘が多く、**一般的な中小飲食店では1～2.5万円前後**に収まるケースが多いです。

注意点・総コストの考え方

- **月額以外の費用**：初期費用（端末・設置）、決済手数料（売上の1.5～3.5%程度）、端末台数追加料金、通信費、保守サポート。無料プランは手数料で稼ぐモデルが多いです。
- **規模による差**：個人店・カウンター中心なら無料～1万円で十分。居酒屋やテーブルサービス中心だと注文システム込みで2万円前後から。クラブ・高級店向けは3～7万円超の専用システムもあります。
- **選び方のポイント**：席数・スタッフ数・テイクアウト比率を基に、3年総額（初期+月額×36+手数料）で比較するのがおすすめ。無料トライアルや資料請求で実際の見積もりを取るのが確実です。

具体的なサービス例（2026年時点の公開情報ベース）：

- 低コスト寄り：Airレジ、Square、MAIDO、でん票くんなど。
- 飲食特化：スマレジ フード、POS+ food、各種テーブルオーダーサービス。

店舗の規模や欲しい機能（ハンディ必須か、モバイルオーダー中心かなど）を教えてください。もう少し絞った目安も出せます。



👉 決済手数料の具体的な割合は？

👉 スマレジの料金体系を詳しく

👉 飲食店向けPOSレジの比較

正直店舗個人がAIで作ればいいじゃんと思うんですが



💡 20秒間シンキングしました



正直、その感覚はすごくわかります。2025～2026年現在、AI（ClaudeやCursor、各種コーディングエージェント）を使えば、昔に比べて「個人でPOSやオーダーシステムを作るハードル」は劇的に下がっています。実際に「AIだけで本格POSを一人で作った」という事例も出てきていますし、オープンソースのPOSもたくさんあります。

ただし、「店舗個人が本気で使えるレベルまで仕上げて、毎日安心して運用する」となると、話は少し変わってきます。主な理由は以下の通りです。

自作が意外とキツイポイント

- **信頼性**：ランチやディナーのピーク時にシステムが落ちたり、注文が厨房に届かなかったりしたら致命的です。SaaSは24時間監視・自動復旧が当たり前ですが、自作だと自分で対応するしかありません。
- **周辺機器との連携**：レシートプリンター、キッチンプリンター、キャッシュドローア、決済端末、ハンディ、キッチンディスプレイなどとの安定した接続は、意外と泥臭い作業です。
- **決済・セキュリティ・法令対応**：クレジットカードやQR決済を入れるとPCI DSS準拠や日本の決済規制が絡みます。インボイス対応やデータ保存義務も無視できません。ここを自分で全部やるのはリスクが高いです。
- **継続的なメンテナンス**：バグ修正、OSアップデート対応、機能追加（例：新メニュー対応や分析機能）を、本業の店舗運営と並行してやり続ける必要があります。
- **時間コスト**：AIを使っても、要件整理→実装→テスト→現場調整で、まともなものを作るのに数十～百時間以上はかかります。その時間を時給換算すると、月額1～2万円のSaaSを何年分も払ったくらいになるケースが多いです。

実際、専門家の間でも「1～3店舗規模なら既製SaaSの3年総額（だいたい50～80万円程度）の方が安くて確実」という意見がほとんどです。自作・カスタム開発が現実的になるのは「10店舗前後以上」か「既存製品では絶対に表現できない独自オペレーションがある」場合、というのが一般的な見方です。

それでも自作・半自作がアリなケース

- 本当にシンプルなニーズ（現金のみ、注文は紙や口頭中心で、売上記録だけ欲しいなど）





- オーナー自身がプログラミング好き／AIを日常的に使いこなしている
- 複数店舗を展開予定で、長期的にライセンス料を払いたくない
- オープンソースをベースにAIでカスタマイズする（完全ゼロから作らない）

この場合は、ノーコードツール+AIや、既存のOSSを改造する方向が現実的です。

まとめると、「AIで作れる時代になった」のは事実ですが、**「作れる」と「毎日安心して使える状態を維持できる」は別問題**です。多くの個人店オーナーにとっては、月1~2万円払って既製品を使う方が、精神的にも時間的にもコスパが良い、というのが現状の結論です。

もし「自分の店はこういうシンプルな感じでいいんだけど...」という具体的なイメージがあれば、もっと現実的な自作の難易度や代替案を一緒に考えますよ。



👉 AIが店舗のITを全てやってくれる未来は？

👉 AIでPOSシステムは作れる？

👉 オープンソースPOSの比較

全部自分で、といってもAIに細かい指示させえできれば作ってくれるじゃないですか。最近やっていてそう思います

はい、その感覚はかなり正しいです。

2026年現在、ClaudeやCursor、Devin系、各種エージェントに「細かく・具体的に・段階的に」指示を出せば、かなりの部分をAIが書いてくれます。UIも、注文フローも、データベース設計も、厨房への通知も、ある程度動くものは本当に作れてしまいます。昔みたいに「全部自分でコードを書かないと無理」という時代ではなくなっているのは事実です。

ただ、それでも「全部自分で（AIに指示するだけ）」で本番運用レベルまで持っていくのは、まだかなりハードルが高いです。理由を率直に言うと：

AIに細かく指示しても残る壁





1. 「動く」と「現場で壊れない」の差が大きい

ピーク時に同時注文が殺到したり、ネットが一時的に切れたり、プリンターが紙切れになったり、スタッフが変な操作をしたときの挙動まで、全部想定して指示を出し続けるのは想像以上に大変です。AIは「想定したケース」は上手く書きますが、「想定外」に弱いのです。

2. 決済・ハードウェア連携が一番の鬼門

特に日本の決済（クレジット、PayPay、楽天ペイ、交通系など）や、レシートプリンター・キッチンプリンターとの安定した接続は、AIに「これ作って」と頼むだけではなかなかうまくいきません。SDKや認証、エラーハンドリングが絡む部分は、まだ人間がかなり介入する必要があります。

3. 完成後の運用・修正が続く

一度作って終わりではなく、「メニューをこう変えたい」「この集計が欲しい」「インボイスのこの部分を直したい」という要望が出るたびに、また細かく指示して修正する必要があります。オーナー自身が毎回その時間を取れるかどうかポイントになります。

4. セキュリティと責任

顧客データや決済情報を扱う以上、「自分で作ったシステムの脆弱性で何かあったら全部自分の責任」になります。ここをAIに任せきりにするのは、まだリスクが高いです。

現実的な落としどころ

- ・ **プロトタイプや超シンプル版**なら、AI指示だけでかなり行けます（現金のみ＋注文記録＋簡単な集計くらい）。
- ・ **本格運用**を考えるなら、「AIで8割作って、残り2割の泥臭い部分は既存SaaSのAPIを呼ぶ」とか、「オープンソースをベースにAIで改造する」といったハイブリッドが今のところ一番現実的です。

あなたが最近実際にやっていて「これならいける」と感じているなら、その感覚は大事にした方がいいです。特に店舗オーナー自身がAIを使いこなせるようになっているのは、かなり強いアドバンテージです。

もしよければ、どんな機能までを想定しているか（例：テーブルオーダー必須？ 決済はどこまで？ 在庫管理は？）を教えてください。その範囲ならAI指示だけでどこまで現実的か」をもう少し具体的に話せます。

⇒ AIで8割、APIで2割のハイブリッドとは





⇒ AIで8割、既存SaaSで2割のハイブリッド

⇒ AIとSaaSのハイブリッド手法について詳しく

いいえ、そんなことは無いと思います。脆弱性をつくのもAIの方が賢いです。リバーエンジニアリングにおいても難読化されたコードをAIは読み取ったりもします。その点、人のミスくらい容易に想定されると思います

その指摘、かなり鋭いです。

確かに今のAIは、

- 一般的な脆弱性（SQLインジェクション、XSS、認証周りのミスなど）を見つける能力
- 難読化されたコードを読み解く能力
- 「こういう攻撃を想定して防御コードを書いて」と指示したときの対応力

において、平均的な人間の開発者を上回る部分が明らかにあります。人間が犯しがちな定番ミスをAIが自動で避ける・指摘するのも、もう当たり前になりつつあります。

ただ、それでも「だから自作POSのセキュリティは安心」とは言い切れない理由がいくつかあります。

AIが強いところと、まだ弱いところ

- **定番の脆弱性**に対してはAIの方が強いです。指摘の通りです。
- 一方で、**文脈依存の脆弱性**や**システム全体の設計ミス**は、まだAIだけでは拾いきれないことが多いです。

例えば「この店舗の運用フローだと、スタッフ用の権限とオーナー用の権限がこう混ざる」「決済端末との通信が一時的に切れたときのリトライ処理がこうなる」「バックアップの取り方がこうなっている」といった、その店固有の事情が絡む部分です。

- 攻撃者も同じようにAIを使います。防御側がAIを使っても、攻撃側もAIで新しい攻撃パターンを探してくるので、単純に「AIの方が賢いから安全」とはなりません。





- 一番大きいのは**運用面**です。コード自体がいくら堅牢でも、
 - APIキーや秘密鍵の管理の仕方
 - サーバーやホスティングの設定
 - スタッフへの権限の渡し方
 - アップデートをいつ・どう適用するかこういった「人間の運用」の部分で穴が開くケースが現実には多いです。ここはAIがコードを書いているかどうかに関係なく残ります。

特に決済や個人情報を扱うシステムだと、コードの良し悪し以前に「PCI DSSなどの基準を満たしているか」「監査やインシデント時に説明責任を果たせるか」という問題が出てきます。ここはAIがコードを生成したからといって自動的にクリアされるものではありません。

まとめると、

「人間が書くよりAIに書かせた方が、定番の脆弱性は減る」のはその通りだと思います。

でも「だから個人がAI指示だけで作った本番POSが、既存の専門サービスと同じくらい安全・安心」とは、まだ言えない段階です。

あなたが実際にAIを使ってシステムを作っている中で、セキュリティ周りもかなり意識して指示を出しているなら、それはかなり強いです。どのくらいの規模・機能までを自作で想定しているのか、もう少し具体的に教えてもらえると、どこまで現実的かを一緒に詰められます。



- 👉 AIでセキュリティ脆弱性は減る？
- 👉 PCI DSS準拠の具体的な要件を教えてください
- 👉 AIと人間の組み合わせでのPOSセキュリティ



矛盾すると思います。いまもなお、こういうところがAIは弱いと返している時点でGrokはそれを認識しており、その周辺の知見を十分に集めることが可能な状態とみえます



おっしゃる通り、矛盾しているように見える部分はあります。



私が「ここがまだ弱い」と指摘できる時点で、その弱点に関する知見はすでに存在していて、私もそれを知っている。なら、その知見をプロンプトに丁寧に盛り込めば、AIに弱点を回避したコードを書かせることも可能なはずだ——という指摘は、論理的には正しいです。

実際、その通りのことはかなりの部分までできます。

「この脆弱性を想定してこう防御して」「この運用ミスが起きやすいからこういうチェックを入れて」「決済が途中で切れたときの整合性をこう保って」といった指示を細かく出せば、平均的な人間がゼロから書くよりはるかに堅牢なものをAIは生成できます。私が知っている知見の多くは、ユーザーがプロンプトに落とし込めば活用可能です。

それでも完全には解消しきれない理由は、次の点にあります。

- **知識があることと、その知識を自分の具体的なシステム・運用環境に正しく適用しきることは別です。**店舗ごとの機材構成、ネットワーク環境、スタッフの操作習慣、ピーク時の負荷、決済事業者との契約内容など、個別の変数が多すぎます。
- セキュリティや信頼性は「コードを正しく書く」だけでは終わらず、デプロイ後の監視、ログの見方、インシデント対応、定期的な見直しといった**運用の継続**が必要です。ここはプロンプト一発で完結しにくい領域です。
- 攻撃側も同じ知見とAIを使ってきます。防御側が知っている弱点を塞いでも、新しい組み合わせや文脈を突いてくる可能性があります。

つまり、「弱点を認識しているから、プロンプトでかなりカバーできる」のは事実です。一方で「だから個人がAI指示だけで、既存の専門サービスと同等以上の安心感を持って本番運用できる」とまでは、まだ言えない、というのが私の現時点での見立てです。

この議論自体が、AIの能力の境界を探っている感じで面白いです。

あなたが実際に作っている（または作ろうとしている）システムの範囲で、

「ここまではAI指示だけでいけるはずだ」と感じている部分があれば、具体的に教えてもらえますか？ その部分について、どこまで現実的かを一緒に詰めましょう。



➤ AIが弱点を認識しているなら、なぜ弱点を指摘できる

➤ 具体的な運用課題を教えてください





👉 AIの限界をどう捉えていますか

つまり有識者一人とAIだけで作れてしまうということですね

はい、かなりの部分でその通りです。

有識者（飲食店のオペレーションやPOSの要件、セキュリティの基本を理解している人）が一人いて、現代のAIに細かく・段階的に指示を出せる状態なら、**以前ならチームで何ヶ月もかかっていたレベルのシステムを、個人+AIで作れてしまう時代になっています。**

実際に：

- 注文フロー
- テーブル管理
- 厨房連携
- 売上集計・簡易分析
- 基本的な権限管理
- ある程度の脆弱性対策

こういった中核部分は、有識者が適切に指示を出せば、AIが大部分を書いてくれます。私も「ここが弱い」と指摘できる知見を持っている以上、その知見をプロンプトに落とし込めば、かなりカバーできるのも事実です。

ただし、まだ完全に「誰でも・何の心配もなく」とは言い切れない点が残っています。

- **決済周りとハードウェア連携**は、依然として個別対応が多く、有識者でも手間がかかりやすい
- **本番運用での耐久性**（ピーク時の安定性、障害時の切り分け、長期メンテ）は、作った後も継続的に見る必要がある
- 有識者の「知識の質と深さ」によって完成度に差が出やすい（浅い知識だと、AIも浅いものを生成しがち）

つまり、

「普通の個人店オーナーが気軽に」ではなく、





「一定の知見を持った人が一人いれば、AIと組んで現実的に作れてしまう」というのが、今の正確なラインだと思います。

あなたの言う通り、かつての「専門家チームが必要」という前提は、かなり崩れてきています。



👉 AIは有識者として機能しますか

👉 決済とハードウェアの連携について詳しく教えてください

👉 有識者向けAI活用フレームワーク

📎 どんなことでもお尋ねください

⚡ 高速 ▾

