

< html >

Moja pierwsza strona internetowa

*WordPress, no-code, AI-buildery, HTML/CSS i Shopify
— jak postawić pierwszą stronę w 2026 roku*

MzdrowY
2026

< /html >

Moja pierwsza strona internetowa

Moja pierwsza strona internetowa

*WordPress, no-code, AI-buildery, HTML/CSS i Shopify — jak
postawić pierwszą stronę w 2026 roku*

MzdrowY

2026

Moja pierwsza strona internetowa

WordPress, no-code, AI-buildery, HTML/CSS i Shopify — jak postawić pierwszą stronę w 2026 roku

Autor: MzdrowY

Copyright © 2026

Pobrano z <https://mzdrowy.github.io/pl>

Wszelkie prawa zastrzeżone.

Spis treści

Spis treści.....	6
Wstęp	7
Rozdział 1. Przegląd rynku — stan vs trend.....	9
Rozdział 2. WordPress.....	13
Rozdział 3. HTML/CSS — jak działa strona.....	22
Rozdział 4. No-code i AI-natywne buildery	31
Rozdział 5. Sklep internetowy — Shopify.....	37
Rozdział 6. Hosting i domena.....	44
Rozdział 7. Co wybrać — podsumowanie	47
Słowniczek pojęć	50

Wstęp

Pamiętasz moment, w którym po raz pierwszy pomyślałeś: „chciałbym mieć swoją stronę internetową”? Może to było przy okazji założenia własnej firmy. Może chciałeś mieć miejsce na portfolio, blog albo prosty sklep z rękodzielniczymi produktami. A może po prostu znudziło Ci się tłumaczenie znajomym, że „strona jest w budowie”, kiedy pytają o Twoją działalność.

I potem zacząłeś szukać informacji. I trafiłeś na ścianę sprzecznych rad. Jeden artykuł mówi, że trzeba uczyć się programowania. Drugi, że WordPress jest przestarzały. Trzeci poleca platformę, o której nigdy nie słyszałeś, a czwarty każe Ci zapłacić specjalistom kilka tysięcy złotych, zanim zrobisz cokolwiek samodzielnie.

Ta książka powstała, żeby uciąć ten chaos.

Nie znajdziesz tu twierdzenia, że istnieje jeden najlepszy sposób na zbudowanie pierwszej strony internetowej. Bo nie istnieje. Jest kilka dobrych sposobów, a właściwy wybór zależy od Twojej konkretnej sytuacji — ile masz czasu, czy planujesz sprzedawać produkty, czy chcesz mieć pełną kontrolę, czy wolisz mieć to po prostu zrobione. Zamiast namawiać Cię na jedno rozwiązanie, ta książka pomoże Ci rozpoznać, które z nich jest Twoje, i poprowadzi Cię przez nie krok po kroku.

Piszę to w połowie 2026 roku i to ma znaczenie. Internet, który opisuję na tych stronach, różni się od tego z poradników sprzed pięciu czy dziesięciu lat — nie dlatego, że stare zasady przestały działać, ale dlatego że pojawiły się nowe, realne opcje. WordPress wciąż napędza prawie połowę stron internetowych na świecie, ale po raz pierwszy w swojej historii systematycznie traci udział na rzecz prostszych, szybszych rozwiązań. Równoległe wyrosła zupełnie nowa kategoria narzędzi — buildery, w których po prostu opisujesz słowami, jak ma wyglądać Twoja strona, a sztuczna

inteligencja generuje ją w ciągu minut. Pominięcie tego tematu w poradniku napisanym dziś byłoby nieuczciwe względem Ciebie jako czytelnika.

Dla kogo jest ta książka? Dla każdego, kto nigdy wcześniej nie zbudował strony internetowej i nie chce w tym celu zapisywać się na kurs programowania trwający pół roku. Nie musisz umieć nic z zakresu informatyki, żeby z niej skorzystać — każdy techniczny termin, który się tu pojawia, wyjaśniam w momencie, w którym jest potrzebny, a na końcu książki znajdziesz też słowniczek, do którego możesz wracać.

Czego się nauczysz? Przejdziemy przez wszystkie realne drogi budowania strony w 2026 roku — WordPress, samodzielne pisanie kodu w HTML i CSS, no-code i AI-buildery oraz Shopify dla sklepów internetowych. Każdej z nich poświęcam tyle uwagi, ile naprawdę zasługuje, opierając się na realnych danych rynkowych, a nie domysłach. Dowiesz się też, jak wybrać domenę i hosting — element wspólny dla większości tych dróg — i jak rozpoznać, czy dobrze wybrałeś, zanim zainwestujesz w to tygodnie pracy.

Jedna rzecz, o którą proszę na wstępie: nie musisz przeczytać tej książki od deski do deski, zanim zaczniesz działać. Rozdział 1 pomoże Ci szybko zorientować się, która droga jest Twoja — a stamtąd możesz przejść prosto do odpowiadającego jej rozdziału. Najlepsza strona internetowa to ta, która w końcu zostaje opublikowana, nie ta, której budowę planujesz w nieskończoność.

Zaczynamy.

Rozdział 1. Przegląd rynku — stan vs trend

1.1 Jak dziś robi się strony internetowe — cztery drogi

Wyobraź sobie, że chcesz mieć dom. Masz kilka opcji. Możesz kupić gotowy projekt i zlecić budowę firmie, która zna się na rzeczy — dostajesz solidny, sprawdzony dom, ale w ramach katalogu, który ona oferuje. Możesz kupić działkę i budować własnymi rękami, cegła po cegle — najwięcej kontroli, ale też najwięcej pracy. Możesz zamówić dom modułowy, który przyjeżdża w częściach i składa się w kilka dni. Albo — to najnowsza opcja — możesz po prostu opisać, jak ma wyglądać Twój dom, a ktoś (albo coś) zbuduje go za Ciebie.

Z budowaniem stron internetowych jest podobnie. W 2026 roku każdy, kto chce postawić swoją pierwszą stronę, wybiera między czterema drogami.

WordPress. To zdecydowanie najpopularniejszy wybór na świecie — według W3Techs, niezależnej firmy badającej technologie internetowe, w czerwcu 2026 roku WordPress napędza prawie 42% wszystkich stron internetowych na świecie. To gotowy system, na który nakładasz motyw (wygląd) i wtyczki (funkcje), trochę jak meble z instrukcją montażu — nie musisz projektować śrubek, ale i tak samodzielnie wszystko składasz.

Liczba 42% pochodzi z W3Techs, które bada około 10 milionów najpopularniejszych stron i liczy każdą domenę (np. mojablog.pl) jako jeden wynik, wliczając subdomeny do głównej. Istnieje też drugie źródło — HTTP Archive — które opiera się na danych z raportu Chrome User Experience (CrUX) i liczy każdą subdomenę osobno. Według niego (dane GravityKit, kwiecień 2026) WordPress ma około 33% rynku. Różnica ~9 punktów procentowych wynika z metodologii, nie z błędu — oba źródła są wiarygodne, po prostu mierzą co innego.

HTML i CSS od podstaw. To budowanie cegła po cegle. Sam piszesz kod, który mówi przeglądarce, co i jak wyświetlić. Daje to pełną kontrolę i jest świetną szkołą zrozumienia, jak właściwie działa internet — ale wymaga najwięcej czasu i nauki, zanim zobaczysz efekt.

No-code i AI-buildery. Tu mieszczą się klasyczne narzędzia typu „przeciągnij i upuść” (Wix, Squarespace, Webflow) oraz ich młodszy, szybko rosnący kuzyni — buildery, w których po prostu opisujesz stronę zwykłym językiem, a sztuczna inteligencja generuje ją za Ciebie (np. Lovable czy Wix AI). To najszybsza droga od pomysłu do działającej strony.

Shopify (i jemu podobne). Jeśli Twoim celem nie jest strona, a sklep internetowy, to osobna kategoria z własnymi regułami — gotowa platforma e-commerce, gdzie płacisz miesięczny abonament i prowizję od sprzedaży, a w zamian dostajesz koszyk, płatności i magazyn działające od pierwszego dnia.

Jest jeszcze jeden element, który dotyczy każdej z tych dróg bez wyjątku: domena (adres Twojej strony) i hosting (miejsce, gdzie ta strona „mieszka” w internecie). To temat rozdziału 6 — niezależnie od tego, którą drogę wybierzesz, w pewnym momencie będziesz musiał kupić działkę i podać adres.

1.2 Stan rynku vs trend: dlaczego „popularne” nie znaczy „warte wyboru dziś”

Tu chciałbym Cię ostrzec przed jedną pułapką, w którą łatwo wpaść, czytając statystyki o internecie.

Kiedy widzisz liczbę „WordPress napędza 42% stron internetowych”, patrzysz na zdjęcie całego istniejącego internetu — łącznie z milionami stron postawionych dziesięć, piętnaście lat temu, które po prostu wciąż działają, bo nikt nie miał powodu ich zmieniać. To w dużej mierze efekt bezwładności, nie codziennego wyboru. Gdyby ktoś, kto postawił taką stronę w 2012 roku, zaczynał dziś od zera — wcale nie jest pewne, że wybrałby to samo rozwiązanie.

A Ty właśnie zaczynasz od zera. Dlatego ważniejsze od pytania „co jest najpopularniejsze” jest pytanie „co wybierają ludzie, którzy zaczynają dziś, tak jak Ty”.

I tu obraz robi się ciekawszy. WordPress, mimo wciąż ogromnego udziału, w 2026 roku po raz pierwszy od 20 lat systematycznie traci udział w rynku — z ponad 43% jeszcze w 2025 roku. W tym samym czasie Wix rośnie o ponad 30% rok do roku, a w branży e-commerce już ponad połowa nowych sklepów internetowych powstałych w 2025 roku wybrała gotowe platformy typu SaaS (Search Engine Journal, dane za 2025), a nie własny, samodzielnie utrzymywany CMS.

Najmłodsza, ale najszybciej rosnąca kategoria to AI-buildery — narzędzia, w których opisujesz stronę słowami. Rynek tych narzędzi w 2026 roku wart jest już ponad 3 miliardy dolarów, a — co dla Ciebie szczególnie istotne — na platformie Hostinger Horizons nawet 93–95% użytkowników to ludzie stawiający swoją pierwszą stronę w życiu. To dokładnie Twoja sytuacja.

Wniosek nie jest taki, że WordPress jest „złym” wyborem — dla wielu sytuacji wciąż jest najlepszy i poświęciłem mu najwięcej stron w tej książce nie bez powodu. Wniosek jest taki, że wybór powinien wynikać z Twojej sytuacji, a nie z tego, co zrobiła większość internetu przez ostatnie dwie dekady. Każdy kolejny rozdział tej książki ma Ci pomóc dopasować drogę do Ciebie — nie odwrotnie.

1.3 Który sposób jest dla Ciebie? Krótki test

Zanim przejdziesz dalej, odpowiedz sobie szczerze na cztery pytania. Nie ma złych odpowiedzi — każda prowadzi Cię do innego rozdziału.

1. Co chcesz osiągnąć?

Jeśli głównie chcesz sprzedawać produkty → przejdź do rozdziału 5 (Shopify).

Jeśli chcesz wizytówki, bloga, portfolio albo strony firmowej → czytaj dalej.

2. Ile czasu chcesz poświęcić na samą naukę, zanim strona będzie gotowa?

Mało, najlepiej dziś → rozdział 4 (no-code i AI-buildery).

Mogę poświęcić kilka weekendów na naukę w zamian za większą kontrolę → rozdział 2 (WordPress).

Chcę naprawdę zrozumieć, jak to wszystko działa, nawet jeśli zajmie to więcej czasu → rozdział 3 (HTML/CSS).

3. Czy zależy Ci na pełnej kontroli i niezależności (własny hosting, brak miesięcznych opłat za platformę), czy bardziej na prostocie i braku martwienia się o szczegóły techniczne?

Kontrola i niezależność → WordPress lub HTML/CSS.

Prostota, nawet za miesięczny abonament → no-code lub AI-buildery.

4. Czy Twoja strona ma rosnąć i rozwijać się przez lata, czy raczej potrzebujesz czegoś „na już”?

Na lata, z myślą o rozwoju → WordPress.

Na już, z możliwością zmiany w przyszłości → no-code/AI-buildery.

Jeśli po tym teście wciąż nie jesteś przekonany — to dobry znak, nie problem. Rozdział 7 zawiera szczegółowe drzewko decyzyjne i tabelę porównawczą wszystkich dróg, do której możesz wrócić po przeczytaniu poszczególnych rozdziałów.

1.4 Jak korzystać z tej książki

Tej książki nie trzeba czytać od deski do deski. Rozdział 1 i 7 (ten, który właśnie czytasz, i podsumowanie na końcu) warto przeczytać zawsze — dają Ci mapę i pomagają sprawdzić, czy dobrze wybrałeś drogę. Pomiędzy nimi wybierz rozdział odpowiadający Twojej drodze z testu powyżej i skup się na nim. Pozostałe rozdziały możesz przeglądać później, kiedy zmienią się Twoje potrzeby — albo kiedy zwyczajnie będziesz ciekawy, jak to wygląda z drugiej strony.

Jedna rzecz, o którą proszę: nie traktuj liczb i statystyk w tej książce jako wyroku. Internet zmienia się szybko — to, co prawdziwe w czerwcu 2026 roku, za dwa lata może wyglądać inaczej. Liczby mają Ci pomóc zrozumieć ogólny kierunek, nie zastąpić Twojej własnej oceny sytuacji.

Idziemy dalej.

Rozdział 2. WordPress

2.1 Co to jest WordPress i dla kogo

WordPress to system do zarządzania treścią — w branży mówi się CMS (Content Management System). Oznacza to, że nie piszesz strony od zera w kodzie, tylko logujesz się do panelu, klikasz „dodaj nową stronę”, wpisujesz tekst, wstawiasz zdjęcia i publikujesz. Cała techniczna „instalacja elektryczna” — to, jak strona jest zapisywana, wyświetlana i obsługiwana — działa w tle, a Ty zajmujesz się tylko tym, co widzi odwiedzający.

Zaczął jako prosty silnik do blogowania w 2003 roku. Dwie dekady później napędza prawie 42% wszystkich stron internetowych na świecie — od małych wizytówek lokalnych firm, przez blogi, po strony dużych mediów i serwisów korporacyjnych.

Dla kogo jest dobrym wyborem? Dla każdego, kto chce mieć dużą kontrolę nad swoją stroną i jej rozwojem, nie chce płacić miesięcznego abonamentu za samą platformę (choć hosting i tak trzeba opłacać) i jest gotów poświęcić trochę czasu na naukę panelu. Jeśli planujesz stronę, która będzie z Tobą rosła przez lata — nowe podstrony, blog, może w przyszłości sklep — WordPress jest jednym z najbezpieczniejszych wyborów właśnie dlatego że tak wiele osób już go używa: każdy problem, na który trafisz, ktoś już rozwiązał i opisał w internecie.

2.2 WordPress.com vs WordPress.org — kluczowa różnica

To pierwsza pułapka, w którą wpada prawie każdy początkujący, bo nazwy są ładząco podobne, a chodzi o dwie różne rzeczy.

WordPress.com to usługa hostowana — coś jak wynajęcie mieszkania z umeblowaniem. Rejestrujesz się, płacisz (albo korzystasz z darmowego planu z ograniczeniami) i dostajesz stronę działającą „pod kluczem”, na serwerach firmy Automattic. Wygodne, ale ograniczone: na niższych planach nie zainstalujesz własnych wtyczek ani motywów, a Twoja strona zawsze będzie częściowo zależna od zasad tej platformy.

WordPress.org to oprogramowanie — darmowy program, który pobierasz i instalujesz na własnym hostingu (o hostingu więcej w rozdziale 6). To jak kupno mieszkania na własność: musisz sam zadbać o instalację i bieżące naprawy, ale masz pełną kontrolę — możesz zainstalować dowolną wtyczkę, dowolny motyw, zmienić dosłownie wszystko.

W tej książce, kiedy mówimy „WordPress”, mamy na myśli wersję .org — to ona daje realną elastyczność i to jej dotyczy reszta tego rozdziału. Jeśli zależy Ci wyłącznie na prostocie i nie planujesz rozbudowy strony w przyszłości, WordPress.com może wystarczyć — ale wtedy równie dobrze warto rozważyć narzędzia z rozdziału 4.

2.3 Instalacja — krok po kroku

Dobra wiadomość: w 2026 roku praktycznie nikt nie instaluje WordPressa „ręcznie”, przez wgranie plików na serwer. Większość firm hostingowych ma instalator jednym kliknięciem. Oto typowa ścieżka:

Od maja 2026 dostępny jest WordPress 7.0. Większość instalatorów hostingowych domyślnie instaluje tę wersję — dostajesz więc od razu natywną integrację z AI (WP AI Client w core), ulepszony edytor i poprawioną wydajność. Wątek AI to jeden z motywów przewodnich tej książki — w WordPress 7.0 masz go od razu po instalacji, bez dodatkowych wtyczek.

- Kupujesz hosting i domenę (rozdział 6 wyjaśnia, jak wybrać). Większość firm hostingowych specjalizujących się w WordPressie oferuje go w cenie pakietu.
- W panelu hostingu szukasz opcji „Zainstaluj WordPress” — zwykle jest to wyraźnie oznaczony przycisk lub ikona, czasem nazwana „Softaculous” albo „Auto Installer”.
- Wypełniasz krótki formularz: nazwa strony, nazwa użytkownika administratora, hasło (użyj menedżera haseł i ustaw coś naprawdę silnego — to pierwsza linia obrony przed włamaniem), adres e-mail.
- Czekasz 2–5 minut, aż system zainstaluje pliki i bazę danych.

- Dostajesz adres panelu administracyjnego — zwykle to Twoja-domena.pl/wp-admin. Zapisz go, będziesz go używać przy każdej wizycie.

To wszystko. Jeśli Twój hosting nie ma takiego instalatora (rzadkość w 2026 roku, ale się zdarza), warto zmienić hosting — ręczna instalacja jest możliwa, ale dla pierwszej strony niepotrzebnie komplikuje życie.

2.4 Pierwsze ustawienia po instalacji

Zaraz po pierwszym logowaniu do panelu warto załatwić kilka rzeczy, zanim zajmiesz się wyglądem strony:

- Ustawienia ogólne (Ustawienia → Ogólne): tytuł strony, krótki opis (tagline), strefa czasowa i język panelu.
- Bezpieczeństwo loginu: zmień nazwę użytkownika z domyślnego „admin” na coś unikalnego, jeśli instalator nie zrobił tego automatycznie — to najczęściej atakowana nazwa konta na świecie.
- Ustawienia → Bez względne linki (Permalinks): zmień strukturę adresów URL na „Nazwa wpisu” — domyślna struktura z datami i znakami zapytania jest nieczytelna i gorsza dla SEO.
- Usuń przykładową treść: WordPress instaluje się z przykładowym wpisem „Witaj, świecie!” i przykładową stroną — usuń je, żeby nie zostały przez przypadek opublikowane.
- Sprawdź wersję WordPressa i włącz automatyczne aktualizacje drobnych poprawek bezpieczeństwa (Ustawienia → Ogólne, sekcja aktualizacji) — to jedna z najważniejszych rzeczy chroniących Cię przed włamaniem.

Te pięć minut na starcie oszczędza potem godziny problemów.

2.5 Wybór motywu — na co zwracać uwagę w 2026 roku

Motyw (theme) to szata graficzna Twojej strony — kolory, układ, typografia, czasem gotowe wzory podstron. W 2026 roku zdecydowanie dominują motywy blokowe (block themes), zaprojektowane do pracy z edytorem Gutenberg, o którym za chwilę.

Na co zwrócić uwagę przy wyborze:

- Lekkość i szybkość. Modny, „ciężki” motyw z dziesiątkami wbudowanych efektów może spowalniać stronę, co szkodzi i użytkownikom, i SEO. Sprawdzone, lekkie

motywy (np. GeneratePress, Kadence, Astra) są w 2026 roku standardem wśród profesjonalistów właśnie z tego powodu.

- Zgodność z edytorem blokowym, nie ze starszym systemem szablonów — to zapewnia łatwiejszą edycję bez kodu.
- Responsywność — motyw musi dobrze wyglądać na telefonie. Sprawdź to w podglądzie przed instalacją, najlepiej na prawdziwym telefonie, nie tylko w podglądzie w panelu.
- Wsparcie i aktualizacje — motywy z dawno nieaktualizowanym wpisem na liście zmian to ryzyko (błędy bezpieczeństwa, niezgodność z nowymi wersjami WordPressa).

Darmowy motyw z oficjalnego katalogu WordPress.org w większości przypadków wystarczy na pierwszą stronę. Płatne motywy (zwykle 150–300 zł jednorazowo) dają więcej gotowych wzorów i wsparcie techniczne — warto rozważyć, jeśli darmowa opcja Ci nie satysfakcjonuje wizualnie.

2.6 Edytor Gutenberg — podstawy

Gutenberg to domyślny edytor treści w WordPressie, w którym budujesz strony i wpisy z bloków — każdy akapit, zdjęcie, przycisk czy nagłówek to osobny blok, który możesz przesuwać, edytować i stylizować niezależnie.

Podstawowa logika pracy:

- Klikasz „+”, żeby dodać nowy blok — masz do wyboru m.in. akapit, nagłówek, obraz, listę, przycisk, kolumny.
- Każdy blok ma własny panel ustawień po prawej stronie — tam zmieniasz kolory, odstępy, wyrównanie.
- Bloki Grupa i Kolumny pozwalają budować bardziej złożone, responsywne układy — np. trzy kolumny z usługami obok siebie, które na telefonie automatycznie ułożą się jedna pod drugą.
- Edytor strony głównej i podstron działa tak samo jak edytor wpisów na blogu — to ta sama logika w obu miejscach.

Najczęstszy błąd początkujących: przesadzanie z liczbą różnych bloków i kolorów na jednej stronie. Trzymaj się prostoty — jedna lub dwie czcionki, ograniczona paleta kolorów, dużo białej przestrzeni. To nie tylko wygląda lepiej, ale też ładuje się szybciej.

Warto poświęcić godzinę na samo „zabawianie się” edytorem na testowej stronie, zanim zaczniesz budować właściwą — szybko poczujesz, jak to działa.

2.7 Strony vs wpisy — kiedy używać czego

To rozróżnienie myli wielu początkujących, a jest proste, gdy raz je zrozumiesz.

Strony (Pages) to treść statyczna, która rzadko się zmienia: „O nas”, „Kontakt”, „Oferta”, strona główna. Mają swoją stałą pozycję w menu i nie są ułożone chronologicznie.

Wpisy (Posts) to treść dynamiczna i chronologiczna: artykuły na blogu, aktualności, ogłoszenia. Domyślnie wyświetlają się od najnowszego do najstarszego i mogą być kategoryzowane oraz tagowane.

Prosta zasada: jeśli treść odpowiada na pytanie „co to jest / kim jesteśmy” — to strona. Jeśli odpowiada na pytanie „co się wydarzyło / co nowego” — to wpis. Wizytówka firmowa bez bloga może obejść się wyłącznie stronami. Jeśli planujesz regularnie publikować treści i budować widoczność w Google poprzez nowe artykuły, wpisy (i blog) są dla Ciebie najważniejsze.

2.8 Menu, widgety, stopka — budowa nawigacji

Sama treść stron nie wystarczy — odwiedzający musi mieć jak się między nimi poruszać.

Menu (Wygląd → Menu) to główna nawigacja, najczęściej widoczna na górze strony. Tworzysz je, dodając do niego wybrane strony w wybranej kolejności, a w motywach blokowych edytujesz je bezpośrednio w edytorze pełnej strony (Wygląd → Edytor).

Widgety to mniejsze bloki funkcjonalne, które możesz umieszczać w określonych miejscach motywu — najczęściej w bocznej kolumnie (sidebar) lub w stopce. Przykłady: lista najnowszych wpisów, formularz kontaktowy, ikony social media.

Stopka (footer) to pas na dole każdej podstrony — zwykle zawiera dane kontaktowe, linki do polityki prywatności i regulaminu (prawnie wymagane w UE), czasem mapę strony. W motywach blokowych edytujesz ją tak samo jak każdą inną część strony, w Edytorze pełnej strony.

Dobra nawigacja ma jedną cechę wspólną niezależnie od strony: użytkownik w trzech kliknięciach powinien dotrzeć do każdej ważnej informacji. Jeśli Twoje menu ma więcej niż 6–7 pozycji głównych, prawdopodobnie warto je pogrupować.

2.9 Najważniejsze wtyczki

Wtyczki (plugins) to dodatki rozszerzające funkcje WordPressa — to one odpowiadają za reputację WordPressa jako systemu „do wszystkiego”. W oficjalnym katalogu jest ich dziesiątki tysięcy, ale na pierwszą stronę potrzebujesz naprawdę niewielu kategorii:

- Formularz kontaktowy — np. WPForms albo Contact Form 7. Bez tego odwiedzający nie ma jak się z Tobą skontaktować inaczej niż przez dane w stopce.
- SEO — np. Yoast SEO albo Rank Math. Prowadzą Cię krok po kroku przez optymalizację każdej strony pod kątem Google (więcej w podrozdziale 2.10).
- Bezpieczeństwo — np. Wordfence. Blokuję podstawowe próby włamań i skanuje stronę pod kątem złośliwego kodu.
- Kopie zapasowe (backup) — np. UpdraftPlus. Automatycznie zapisuje kopię Twojej strony, żebyś mógł ją przywrócić, jeśli coś pójdzie nie tak.
- Pamięć podręczna / wydajność (cache) — np. WP Rocket (płatna) albo WP Super Cache (darmowa). Przyspiesza wczytywanie strony, co ma znaczenie i dla użytkowników, i dla SEO.

Złota zasada dotycząca wtyczek: instaluj tylko te, których faktycznie potrzebujesz, i regularnie usuwaj nieużywane. Każda dodatkowa wtyczka to potencjalne dodatkowe ryzyko bezpieczeństwa i obciążenie strony — dziesięć zainstalowanych „na wszelki wypadek” wtyczek szkodzi bardziej niż pomaga.

2.10 Podstawy SEO w WordPressie

SEO (Search Engine Optimization) to zestaw działań, które pomagają Twojej stronie pojawiać się wyżej w wynikach wyszukiwania Google. To temat na osobną książkę, ale kilka podstaw warto znać od pierwszego dnia:

- Każda strona i wpis powinien mieć unikalny tytuł i opis meta — to właśnie tym zarządza wtyczka SEO z poprzedniego podrozdziału, prowadząc Cię przez pola tytułu i opisu pod każdym edytowanym tekstem.
- Nagłówki mają hierarchię — jeden nagłówek główny (H1) na stronę, pod nim nagłówki H2, pod nimi H3. Edytor blokowy pozwala to łatwo ustawić przy każdym bloku nagłówka.

- Szybkość strony ma znaczenie — stąd wtyczka cache z poprzedniego podrozdziału i lekki motyw z 2.5.
- Treść pisana dla ludzi, nie dla algorytmu — Google w 2026 roku coraz silniej premiuje rzeczywistą wartość i ekspertyzę tekstu, a nie sztuczne upychanie słów kluczowych.
- Każda strona dostępna z menu — strony „osierocone”, do których nigdzie nie prowadzi link, są trudniejsze do znalezienia przez wyszukiwarki.

Na start tych pięć zasad wystarczy. Reszta — budowanie linków, analiza słów kluczowych, content marketing — to już praca na kolejne miesiące działania strony, nie na pierwszy dzień.

2.11 Bezpieczeństwo i backupy

WordPress, właśnie dlatego że jest tak popularny, jest też najczęściej atakowanym celem botów skanujących internet w poszukiwaniu słabych haseł i nieaktualnych wersji. To nie powód do paniki — to powód do kilku prostych nawyków:

- Aktualizuj — WordPressa, motyw i wszystkie wtyczki, najlepiej automatycznie dla drobnych poprawek. Większość włamań wykorzystuje znane, już załatane błędy w nieaktualnym oprogramowaniu.
- Silne, unikalne hasło i, jeśli hosting to umożliwiał, weryfikacja dwuetapowa przy logowaniu do panelu.
- Regularne kopie zapasowe poza serwerem, na którym stoi strona (np. automatyczny zapis na Google Drive czy Dropbox przez wtyczkę backupową) — jeśli coś pójdzie nie tak, przywracasz stronę w kilka minut, zamiast budować ją od zera.
- Ograniczona liczba kont administratora — każde konto z dostępem to potencjalny słaby punkt.

Pomyśl o tym jak o zamykaniu drzwi na klucz: nic z tego nie gwarantuje stuprocentowego bezpieczeństwa, ale eliminuje 95% potencjalnych problemów przy minimalnym nakładzie pracy.

2.12 Najczęstsze błędy początkujących

Z doświadczenia tysięcy osób stawiających swoją pierwszą stronę na WordPressie, kilka pułapek powtarza się najczęściej:

- Instalowanie zbyt wielu wtyczek „na wszelki wypadek” — spowalnia stronę i zwiększa ryzyko konfliktów między wtyczkami.

- Ignorowanie aktualizacji miesiącami, aż w końcu coś się psuje albo strona zostaje zhakowana.
- Brak backupów — aż do dnia, w którym coś idzie nie tak i nie ma czego przywrócić.
- Przesadzanie z efektami wizualnymi — animacje, sliderzy, jaskrawe kolory na każdej sekcji — które wyglądają „ciekawie” w edytorze, a realnie rozpraszą i spowalniają stronę.
- Pisanie treści dla siebie, nie dla odwiedzającego — strona firmowa pełna ogólników o „pasji i profesjonalizmie”, bez konkretnej informacji, co dokładnie robicie i jak się z Wami skontaktować.
- Brak testu na telefonie przed publikacją — większość odwiedzających zobaczy Twoją stronę najpierw na małym ekranie.

Żaden z tych błędów nie jest katastrofalny, jeśli go znajdziesz i poprawisz wcześniej — gorzej, gdy zostaje niezauważony miesiącami.

2.13 Ile to kosztuje — realny budżet

WordPress jako oprogramowanie jest darmowy, ale strona na WordPressie nigdy nie jest „za darmo”. Realny budżet na pierwszy rok wygląda orientacyjnie tak:

- Hosting: od około 100–150 zł rocznie za podstawowy pakiet współdzielony, do kilkuset złotych miesięcznie za hosting dedykowany WordPressowi przy większym ruchu.
- Domena: 30–80 zł rocznie za popularne rozszerzenia (.pl, .com).
- Motyw: 0 zł (darmowy z katalogu) do 300–400 zł jednorazowo za motyw premium.
- Wtyczki premium (opcjonalnie): pojedyncze wtyczki płatne kosztują zwykle 100–300 zł rocznie, ale na start większość potrzebnych funkcji pokrywają darmowe wersje.

Realnie pierwsza skromna strona wizytówkowa na WordPressie kosztuje od około 150 do 500 zł w pierwszym roku, jeśli budujesz ją samodzielnie. To bez kosztów ewentualnego zlecenia pracy specjalistom — które, w zależności od zakresu, zaczynają się zwykle od kilkuset i sięgają kilku tysięcy złotych.

2.14 Kiedy WordPress NIE jest dobrym wyborem

Mimo wszystkich zalet, WordPress nie jest uniwersalną odpowiedzią na każdą potrzebę. Rozważ inną drogę, jeśli:

- Potrzebujesz strony „na już”, w godzinę, i nie planujesz jej rozwijać — no-code lub AI-builder z rozdziału 4 da Ci to szybciej i z mniejszym wysiłkiem.
- Twoim głównym celem jest sklep internetowy z dużą skalą sprzedaży — Shopify (rozdział 5) jest zaprojektowany specjalnie pod e-commerce i odciąży Cię z wielu technicznych decyzji, którymi w WordPressie musiałbyś zarządzać sam przez wtyczkę WooCommerce.
- Chcesz dogłębnie zrozumieć, jak działa internet, a nie tylko mieć działającą stronę — wtedy warto zacząć od HTML i CSS z rozdziału 3, nawet jeśli później i tak przejdziesz na WordPress.
- Absolutnie nie chcesz zajmować się aktualizacjami, bezpieczeństwem i backupami samodzielnie — choć istnieje hosting zarządzany, który robi to za Ciebie (za wyższą cenę), to wciąż wymaga większej uwagi niż w pełni hostowane platformy z rozdziału 4.

WordPress jest jak dobrze wyposażony warsztat — daje narzędzia do zbudowania prawie wszystkiego, ale to wciąż Ty musisz nauczyć się nimi posługiwać. Dla wielu osób to inwestycja, która się opłaca. Dla innych — niepotrzebny ciężar, kiedy prostsze rozwiązanie wystarczyłoby równie dobrze.

Rozdział 3. HTML/CSS — jak działa strona

3.1 Jak przeglądarka czyta stronę — model mentalny

Zanim napiszesz pierwszą linijkę kodu, warto zrozumieć, co właściwie dzieje się między wpisaniem adresu w przeglądarce a zobaczeniem strony na ekranie. To prostszy proces, niż się wydaje.

Kiedy wpisujesz adres strony, Twoja przeglądarka wysyła prośbę do serwera, na którym ta strona „mieszka”: „daj mi swoje pliki”. Serwer odsyła z powrotem zazwyczaj trzy rodzaje plików:

HTML — szkielet i treść strony: jakie elementy są na stronie i w jakiej kolejności (nagłówek, akapit, zdjęcie, przycisk).

CSS — wygląd: kolory, czcionki, odstępy, układ elementów względem siebie.

JavaScript — zachowanie i interaktywność: co się dzieje, kiedy klikniesz przycisk albo przewiniesz stronę.

Przeglądarka czyta HTML od góry do dołu i buduje z niego coś jak drzewo genealogiczne elementów — w branży nazywane DOM. Potem „maluje” ten szkielet zgodnie z regułami z CSS. Na końcu, jeśli jest jakiś JavaScript, przeglądarka go wykonuje, dodając interaktywność.

W tym rozdziale skupimy się na pierwszych dwóch warstwach — HTML i CSS — bo to one wystarczą, żeby zbudować kompletną, działającą stronę statyczną. Strona „statyczna” oznacza po prostu: te same pliki są wysyłane do każdego odwiedzającego, bez bazy danych i bez logowania się do panelu administracyjnego, jak w WordPressie. To najprostsza możliwa forma strony internetowej — i dobry punkt startowy do zrozumienia wszystkiego, co dzieje się „pod maską” w pozostałych rozdziałach tej książki.

3.2 HTML — znaczniki i struktura dokumentu

HTML (HyperText Markup Language) opisuje treść za pomocą znaczników (tagów) — słów w ostrych nawiasach, które mówią przeglądarce, czym jest dany fragment treści. Większość znaczników występuje w parach: znacznik otwierający i zamykający, z treścią między nimi.

```
<p>To jest akapit tekstu.</p>
```

Każda strona HTML ma stały szkielet:

```
<!DOCTYPE html>
<html lang="pl">
<head>
  <meta charset="UTF-8">
  <title>Tytuł strony w karcie przeglądarki</title>
</head>
<body>
  <h1>To jest nagłówek główny</h1>
  <p>To jest akapit tekstu pod nagłówkiem.</p>
</body>
</html>
```

Kilka elementów tego szkieletu zawsze wygląda tak samo: `<!DOCTYPE html>` informuje przeglądarke, że ma do czynienia ze współczesnym HTML-em. Sekcja `<head>` zawiera informacje „dla przeglądarki i wyszukiwarek”, niewidoczne na samej stronie. Sekcja `<body>` zawiera wszystko, co faktycznie widzi odwiedzający.

Najczęściej używane znaczniki, które wystarczą na dobry początek:

Znacznik	Do czego służy
<code><h1></code> do <code><h6></code>	nagłówki, od najważniejszego do najmniej ważnego
<code><p></code>	akapit tekstu
<code></code>	link do innej strony
<code></code>	obraz
<code></code> i <code></code>	lista punktowana
<code><div></code>	uniwersalny kontener, „pudełko” do grupowania elementów

W 2026 roku standardem jest też HTML semantyczny — używanie znaczników, które same opisują znaczenie fragmentu strony, zamiast samych ogólnych `<div>`-ów: `<header>` na górny pasek, `<nav>` na menu, `<main>` na główną treść, `<section>` na sekcję tematyczną, `<footer>` na stopkę. Dzięki temu strona jest lepiej rozumiana przez wyszukiwarki i czytniki ekranu dla osób niedowidzących — a Tobie, jako autorowi, kod jest po prostu łatwiejszy do ogarnięcia.

3.3 CSS — stylizowanie elementów

Sam HTML daje stronę, która działa, ale wygląda jak dokument tekstowy z lat dziewięćdziesiątych — czarny tekst na białym tle, bez żadnego stylu. CSS (Cascading Style Sheets) to język, który mówi przeglądarce, jak ten szkielet ma wyglądać.

Podstawowa składnia CSS wygląda tak:

```
selektor {
  właściwość: wartość;
}
```

Na przykład:

```
h1 {
  color: navy;
  font-size: 32px;
}
```

To mówi przeglądarce: „każdy nagłówek <h1> pokoloruj na granatowo i ustaw rozmiar czcionki na 32 piksele”.

Żeby stylizować pojedyncze elementy, a nie wszystkie tego typu, używa się klas — etykiet, które sam nadajesz w HTML:

```
<p class="ostrzezenie">Uwaga, to ważna informacja.</p>
```

... a w CSS:

```
.ostrzezenie {
  color: red;
  font-weight: bold;
}
```

Zwróć uwagę na kropkę przed nazwą klasy w CSS — to ona mówi „szukaj elementu z taką klasą”, w odróżnieniu od nazwy samego znacznika.

CSS możesz dołączyć do strony trzema sposobami: bezpośrednio w znaczniku (`style="..."`), w sekcji <head> w bloku <style>, albo — najlepsza i najczęściej stosowana praktyka — w osobnym pliku .css, podłączonym do HTML jedną linią:

```
<link rel="stylesheet" href="styl.css">
```

Osobny plik CSS oznacza, że jedną zmianę kolorów stosujesz do wszystkich podstron strony naraz, zamiast poprawiać każdą z osobna — to jedna z najważniejszych zalet oddzielenia treści (HTML) od wyglądu (CSS).

3.4 Box model i flexbox — podstawy układu

Każdy element na stronie — akapit, zdjęcie, przycisk — przeglądarka traktuje jak prostokątne „pudełko”. Zrozumienie, z czego składa się to pudełko, to jeden z tych momentów, po którym CSS zaczyna mieć sens.

Każde pudełko ma cztery warstwy, od środka na zewnątrz:

- Content — sama treść (tekst, obraz).
- Padding — wewnętrzny odstęp między treścią a krawędzią pudełka.
- Border — obramowanie, jeśli je ustawisz.
- Margin — zewnętrzny odstęp, oddzielający to pudełko od sąsiednich elementów.

```
.karta {
  padding: 16px;
  border: 1px solid #ddd;
  margin: 24px;
}
```

To tłumaczy, dlaczego dwa elementy czasem „dotykają się”, a czasem mają ładny odstęp — to właśnie kwestia paddingu i marginesu.

Kolejny istotny temat to flexbox — narzędzie do układania wielu pudełek względem siebie, np. w rzędku obok siebie albo równo rozłożonych w kolumnie. Ustawia się je na rodzicu, który ma w sobie kilka elementów:

```
.menu {
  display: flex;
  justify-content: space-between;
  align-items: center;
}
```

Ten jeden fragment kodu mówi: „rozłóż elementy wewnątrz w jednym rzędku (`display: flex`), rozsuń je równomiernie na całą szerokość (`justify-content`), i wyśrodkuj je w pionie względem siebie (`align-items`)”. To dokładnie ten typ kodu, który stoi za większością nawigacji na górze stron internetowych.

Flexbox układa elementy w jednym rzędzie albo w jednej kolumnie — działa w jedną stronę. CSS Grid to uzupełnienie: obsługuje dwie strony naraz, czyli wiersze i kolumny jednocześnie, jak tabela w arkuszu kalkulacyjnym. Którego użyć? Flexboxa, gdy elementy mają się ułożyć w linii. Grida, gdy tworzysz szachownicę — siatkę wierszy i kolumn. Na początek przygod z CSS flexbox wystarczy w większości sytuacji, ale warto wiedzieć, że Grid czeka w gotowości, gdy trafisz na układ dwuwymiarowy.

Nie musisz zapamiętywać tych właściwości — większość osób piszących CSS sprawdza je w dokumentacji albo ściądże, dopóki nie wejdą w nawyk. Ważniejsze jest zrozumienie samej logiki: pudełka w pudełkach, z odstępami i układem, które kontrolujesz przez CSS.

3.5 Responsywność — strona na telefonie

Większość odwiedzających zobaczy Twoją stronę najpierw na telefonie, nie na komputerze. Strona, która nie dostosowuje się do małego ekranu, to dziś jeden z najszybszych sposobów na zniechęcenie odwiedzającego.

Pierwszy, niezbędny krok to jedna linia w sekcji <head>:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Bez tego przeglądarki mobilne domyślnie wyświetlają stronę „zmniejszoną”, jakby to był pełny widok desktopowy w miniaturze — dokładnie ten irytujący efekt, który zmusza do powiększania tekstu palcami.

Drugi element to media queries — reguły CSS, które stosują się tylko przy określonej szerokości ekranu:

```
.kontener {
  display: flex;
}
@media (max-width: 600px) {
  .kontener {
    flex-direction: column;
  }
}
```

Ten przykład mówi: „normalnie ułóż elementy w rzędzie, ale jeśli ekran jest węższy niż 600 pikseli, ułóż je jeden pod drugim”. Dzięki temu trzy kolumny obok siebie na komputerze stają się czytelną, pojedynczą kolumną na telefonie.

Dobra praktyka w 2026 roku to myślenie mobile-first — projektowanie i pisanie stylów najpierw dla wąskiego ekranu, a potem dodawanie reguł rozszerzających układ na większe ekrany, a nie odwrotnie. Niezależnie od podejścia: zawsze testuj stronę na prawdziwym telefonie przed publikacją, nie tylko w podglądzie na komputerze.

3.6 Narzędzia developerskie w przeglądarce

Każda współczesna przeglądarka ma wbudowany zestaw narzędzi do podglądu i debugowania kodu strony — i nie trzeba niczego instalować, żeby z nich skorzystać.

Otwiera się je klawiszem F12 (albo prawym przyciskiem myszy na elemencie strony → „Zbadaj element” / „Inspect”). Dwie zakładki będą Ci najbardziej przydatne na początek:

- **Elements** — pokazuje żywy kod HTML strony i, kiedy klikniesz na element, dokładnie jakie reguły CSS na niego działają. Możesz tu na chwilę zmienić wartość (np. kolor czy odstęp) prosto w przeglądarce i zobaczyć efekt, bez zapisywania pliku — świetne do szybkiego eksperymentowania.
- **Console** — pokazuje błędy w kodzie, np. kiedy plik CSS się nie wczytał albo jest literówka w nazwie klasy.

Praktyczna umiejętność, którą warto sobie wyrobić od razu: kiedy coś na stronie „nie wygląda tak, jak powinno”, zamiast zgadywać, kliknij na ten element prawym przyciskiem, wybierz „Zbadaj” i sprawdź, jakie style faktycznie są na niego nałożone. Bardzo często okazuje się, że jakaś inna reguła CSS „nadpisuje” tę, którą próbujesz zastosować — i dev tools pokazują to jaśnie, zamiast zostawiać Cię ze zgadywaniem.

3.7 Projekt: Twoja pierwsza statyczna strona od zera

Czas połączyć to wszystko w jeden, kompletny projekt — prostą, jednostronicową wizytówkę z nagłówkiem, sekcją „o mnie”, listą usług i stopką z danymi kontaktowymi.

Struktura HTML (plik index.html):

```
<!DOCTYPE html>
<html lang="pl">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Jan Kowalski – Fotograf</title>
  <link rel="stylesheet" href="styl.css">
</head>
<body>
  <header class="naglowek">
    <h1>Jan Kowalski</h1>
    <nav>
      <a href="#o-mnie">O mnie</a>
      <a href="#uslugi">Usługi</a>
      <a href="#kontakt">Kontakt</a>
    </nav>
  </header>

  <main>
    <section id="o-mnie" class="sekcja">
```

```
    <h2>O mnie</h2>
    <p>Fotografuję od ponad dziesięciu lat, specjalizując się w sesjach ślubnych
i portretowych.</p>
  </section>
  <section id="uslugi" class="sekcja">
    <h2>Usługi</h2>
    <ul>
      <li>Sesje ślubne</li>
      <li>Portrety indywidualne</li>
      <li>Sesje rodzinne</li>
    </ul>
  </section>
</main>

<footer id="kontakt" class="stopka">
  <p>Kontakt: jan@example.pl | 600 000 000</p>
</footer>
</body>
</html>
```

Style (plik styl.css):

```
body {
  font-family: Arial, sans-serif;
  margin: 0;
  color: #222;
}

.naglowek {
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 16px 24px;
  background-color: #2c3e50;
  color: white;
}

.naglowek nav a {
  color: white;
  margin-left: 16px;
  text-decoration: none;
}

.sekcja {
  padding: 40px 24px;
  max-width: 700px;
}
```

```

margin: 0 auto;
}

.stopka {
background-color: #f4f4f4;
text-align: center;
padding: 16px;
}

@media (max-width: 600px) {
.naglowek {
flex-direction: column;
text-align: center;
}
.naglowek nav a {
margin: 4px 8px;
}
}

```

Zapisz oba pliki w jednym folderze i otwórz `index.html` dwuklikiem — to wszystko, czego potrzeba, żeby zobaczyć efekt w przeglądarce, bez żadnego serwera czy instalacji. Spróbuj zmienić kolory, dodać swoje zdjęcie (znacznik ``), albo dopisać kolejną sekcję — to najlepszy sposób na naukę: psuć i naprawiać prawdziwy kod.

Kiedy będziesz zadowolony z efektu, możesz opublikować taką stronę całkowicie bezpłatnie przez GitHub Pages — opisujemy to dokładnie w rozdziale 6.

3.8 Po co Ci to, jeśli i tak wybierzesz WordPress albo Wix

Jeśli po przeczytaniu tego rozdziału wciąż planujesz zbudować swoją stronę na WordPressie albo w no-code builderze — to nie był czas stracony i to nie przypadek, że ten rozdział jest w tej książce niezależnie od Twojej ostatecznej decyzji.

Każdy motyw WordPressa, każdy szablon Wixa, każda strona wygenerowana przez AI-builder — to w finalnej wersji wciąż HTML i CSS, tylko wygenerowany za Ciebie. Rozumiejąc te podstawy:

- Łatwiej naprawisz drobne problemy — przesunięty element, zły kolor, tekst, który nie chce się zmieścić — bez czekania na pomoc specjalisty.
- Lepiej rozmawiasz z osobami, które zatrudnisz do pracy nad Twoją stroną — rozumiesz, o czym mówią, kiedy wspominają „padding” czy „media query”.

- Świadomie korzystasz z opcji „własny CSS”, którą mają prawie wszystkie buildery z rozdziału 4, do drobnych poprawek wykraczających poza standardowe ustawienia.
- Rozumiesz, dlaczego niektóre rzeczy są trudniejsze niż się wydają – np. dlaczego idealne wyrównanie trzech elementów w rzędku na każdym rozmiarze ekranu to nie kwestia „jednego kliknięcia”, a kompromisu między różnymi szerokościami.

Innymi słowy: nawet jeśli ostatecznie zbudujesz stronę bez napisania ani jednej linii kodu, ten rozdział daje Ci coś, czego żaden builder nie da – zrozumienie, co właściwie dzieje się pod powierzchnią Twojej strony.

Rozdział 4. No-code i AI-natywne buildery

4.1 Mapa terenu: drag-and-drop vs AI-natywne

W tym rozdziale mieszczą się dwie generacje narzędzi, które łączy jedno: żadne z nich nie wymaga pisania kodu.

Klasyczne buildery typu „przeciągnij i upuść” — Wix, Squarespace, Webflow — działają na zasadzie wizualnego edytora: wybierasz element z panelu (zdjęcie, tekst, przycisk), przeciągasz go na stronę, ustawiasz jego wygląd klikając w opcje, a nie pisząc CSS. To wciąż Ty układasz każdy element.

AI-natywne buildery — Lovable, Wix AI, Framer AI i im podobne — to nowsza kategoria, w której opisujesz słowami, co chcesz zobaczyć, a narzędzie generuje stronę za Ciebie. Nie przeciągasz elementów — prowadzisz rozmowę.

Granica między tymi grupami zaciera się z miesiąca na miesiąc — klasyczne platformy dodają własne funkcje AI, a AI-natywne narzędzia dają coraz więcej wizualnej kontroli po wygenerowaniu projektu. W tym rozdziale przejdziemy przez najważniejszych przedstawicieli obu generacji, żebyś mógł wybrać tę, która odpowiada Twojemu stylowi pracy.

4.2 Wix — dla kogo i jak zacząć

Wix to najpopularniejszy klasyczny builder na świecie i jeden z najszybciej rosnących pod względem udziału w nowo powstających stronach. Jego siła to ogromna biblioteka szablonów i edytor, w którym praktycznie nie da się „zepsuć” całej strony jednym kliknięciem.

Dla kogo: dla osób, które chcą mieć w pełni działającą, ładnie wyglądającą stronę w ciągu jednego popołudnia, bez nauki czegokolwiek wcześniej, i którym nie przeszkadza miesięczny abonament w zamian za tę wygodę.

Jak zacząć:

- Rejestrujesz się na wix.com i wybierasz, czy chcesz zacząć od szablonu, czy skorzystać z wbudowanego AI (Wix odpowiada na kilka pytań o Twoją branżę i cele, po czym proponuje gotowy szkic strony).
- Edytujesz wybrany projekt — klikasz na każdy element, by zmienić tekst, zdjęcie, kolor czy odstęp.
- Dodajesz własne podstrony z górnego menu „Strony”, jeśli potrzebujesz więcej niż jednej.
- Publikujesz jednym przyciskiem. Na darmowym planie strona działa na subdomenie Wixa i z reklamami Wixa — żeby podłączyć własną domenę i zdjąć reklamy, potrzebny jest płatny plan.

Wix dobrze sprawdza się przy wizytówkach, portfolio i mniejszych sklepach. Słabiej — jeśli planujesz bardzo nietypowy, niestandardowy układ strony albo migrację w przyszłości na inną platformę, bo eksport poza Wixa jest mocno ograniczony.

4.3 Squarespace — dla kogo i jak zacząć

Squarespace ma węższą bibliotekę szablonów niż Wix, ale wszystkie są zaprojektowane z dużą dbałością o estetykę — to platforma szczególnie popularna wśród fotografów, projektantów, artystów i małych butików, dla których wizualny efekt „od razu profesjonalny” ma duże znaczenie.

Dla kogo: dla osób, którym zależy na elegantszym, bardziej dopracowanym domyślnym wyglądzie niż w przeciętnym darmowym szablonie Wixa, i które są gotowe na odrobinę bardziej stonowaną, mniej elastyczną edycję w zamian za ten efekt.

Jak zacząć:

- Wybierasz szablon dopasowany do branży — biblioteka jest mniejsza, ale dobrze skategoryzowana.
- Edytujesz stronę sekcja po sekcji — Squarespace prowadzi Cię przez gotowe „bloki” treści (zdjęcie z tekstem, galeria, formularz), które łatwo przestawiać, ale trudniej całkowicie przeprojektować.

- Konfigurujesz podstawowe ustawienia SEO i połączenia z mediami społecznościowymi w panelu ustawień.
- Publikujesz i podłączasz domenę — Squarespace nie ma darmowego planu z reklamami jak Wix, każdy plan jest płatny od początku, ale bez wymuszonego brandingu na stronie.

Squarespace dobrze sprawdzi się jako wizytówka portfolio lub mały sklep z dopracowanym designem. Mniej elastyczny niż Wix, jeśli chcesz bardzo nietypowego układu strony.

4.4 Webflow — kiedy warto wejść głębiej

Webflow to inna kategoria niż Wix i Squarespace, mimo że formalnie też jest „builderem bez kodu”. To narzędzie zaprojektowane dla osób, które chcą mieć kontrolę na poziomie zbliżonym do pisania własnego CSS — tylko wizualnie, klikając, a nie pisząc.

Jeśli przeczytałeś rozdział 3, poczujesz się tu niemal jak w domu: Webflow operuje dosłownie na tych samych pojęciach — box model, flexbox, media queries — tylko przez panel ustawień, nie przez kod. To sprawia, że ma znacznie większą krzywą uczenia niż Wix, ale też dużo większy „strop” możliwości — Webflow jest standardem w agencjach projektujących bardziej wymagające, animowane strony marketingowe.

Dla kogo: dla osób, które przeszły przez podstawy z rozdziału 3 i chcą wizualnego interfejsu zamiast pisania kodu od zera, albo dla tych, którzy planują bardziej rozbudowaną, unikalną stronę i są gotowi poświęcić więcej czasu na naukę narzędzia.

Webflow jest droższy niż Wix i Squarespace na porównywalnych planach, ale daje też możliwość eksportu czystego kodu HTML/CSS w niektórych planach — co znacząco zmniejsza ryzyko „uwięzienia” na platformie w porównaniu z Wixem.

Jeśli czujesz, że ten opis brzmi jak „za dużo” na pierwszą stronę — prawdopodobnie masz rację. Webflow to dobry wybór na drugą albo trzecią stronę, nie na pierwszy projekt.

4.5 AI-buildery — opisujesz, AI tworzy

To najmłodsza i najszybciej rosnąca kategoria w tej książce — i, jak wspomnieliśmy w rozdziale 1, skierowana właśnie do osób stawiających swoją pierwszą stronę. Należą do niej narzędzia jak Lovable, Wix AI (czatowa wersja Wixa, odrębna od jego klasycznego edytora) czy Framer AI.

Zasada działania jest podobna we wszystkich: piszesz w zwykłym języku, czego potrzebujesz — np. „strona dla mojej pracowni jogi, z grafiką zajęć, formularzem zapisu i sekcją o instruktorach, w spokojnej, naturalnej kolorystyce” — a narzędzie w ciągu sekund lub minut generuje gotowy szkic całej strony.

Typowy przebieg pracy:

- Opisujesz stronę, najlepiej możliwie konkretnie — im więcej szczegółów (branża, sekcje, ton, przykładowe treści), tym lepszy pierwszy szkic.
- Przeglądasz wygenerowaną wersję i poprawiasz ją kolejnymi wiadomościami w czacie: „zmień kolor główny na zielony”, „dodaj sekcję z opiniami klientów”, „skróć tekst w nagłówku”.
- Po kilku rundach poprawek dopracowujesz drobne szczegóły, czasem już w prostym wizualnym edytorze dołączonym do narzędzia.
- Publikujesz — w zależności od platformy, na własnej domenie od razu albo eksportując gotowy kod.

Tu jest istotna różnica techniczna, o której warto wiedzieć przed wyborem konkretnego narzędzia: niektóre AI-buildery (np. Lovable) generują prawdziwy, eksportowalny kod (najczęściej w technologii React), który możesz zabrać i hostować gdziekolwiek. Inne (np. czatowa wersja Wixa) tworzą stronę zamkniętą w ramach własnej platformy, bez możliwości pełnego eksportu — podobnie jak klasyczny Wix.

Jedna uwaga praktyczna: jakość pierwszego wygenerowanego szkicu bywa nierówna — czasem trafiasz od razu w dobry projekt, czasem dostajesz coś bardzo ogólnego, co wymaga kilku rund poprawek, żeby przestało wyglądać „jak wszystkie inne”. To normalne i wciąż szybsze niż budowanie od zera — ale nie traktuj pierwszego wyniku jako ostatecznego.

4.6 Plusy i minusy: vendor lock-in, koszty, własność danych

Przed podjęciem decyzji warto rozważyć trzy rzeczy, które łatwo przeoczyć w pierwszym entuzjastycznie szybkim efekcie.

Vendor lock-in (uwięzienie u dostawcy) — to pytanie, czy w przyszłości łatwo zabierzesz swoją stronę gdzie indziej. Klasyczne buildery (Wix, Squarespace) w większości nie pozwalają na pełny eksport — Twoja strona „żyje” wewnątrz ich platformy i jeśli zechcesz zmienić dostawcę, w praktyce budujesz nową stronę od zera. Webflow i niektóre AI-buildery dają realny eksport kodu, co znacząco zmniejsza to ryzyko.

Koszty w czasie — w przeciwieństwie do WordPressa, gdzie po opłaceniu hostingu i domeny nie musisz płacić za samo oprogramowanie, buildery z tego rozdziału to zazwyczaj stały, comiesięczny lub roczny abonament, który płacisz tak długo, jak długo strona ma być online — i zwykle rośnie, gdy potrzebujesz więcej funkcji (sklep, większy ruch, usunięcie branding).

Własność danych — jeśli platforma zmieni cennik, ograniczy funkcje albo (rzadko, ale się zdarza) zniknie z rynku, Twoja strona zniknie z nią, o ile nie miałaś możliwości eksportu. Przed wyborem konkretnego narzędzia, szczególnie nowszego AI-buildera, warto poświęcić pięć minut na sprawdzenie w jego dokumentacji, czy i jak eksportuje dane.

Żadna z tych rzeczy nie jest powodem, by unikać tej kategorii narzędzi — dla wielu osób wygoda i szybkość są wystarczająco wartościowe, by zaakceptować te kompromisy. Ważne, żeby był to świadomy wybór, nie niespodzianka odkryta za rok.

4.7 Który builder wybrać — checklista

Krótką checklista, którą warto przejść przed rejestracją w którymkolwiek z tych narzędzi:

- Ile mam czasu na pierwszy efekt? Minuty → AI-builder. Godziny → Wix lub Squarespace. Dni/tygodnie z dużą kontrolą → Webflow.
- Czy zależy mi na unikalnym, niestandardowym designie, czy wystarczy mi solidny, gotowy szablon? Gotowy szablon → Wix/Squarespace. Unikalny układ → Webflow lub AI-builder z dobrym promptem.
- Czy mam jasną wizję strony, czy łatwiej mi ją opisać słowami niż poskładać wizualnie? Łatwiej opisać → AI-builder. Łatwiej zobaczyć i poklikać → klasyczny builder.
- Czy w przyszłości mogę chcieć przenieść stronę na inną platformę albo własny hosting? Tak, to ważne → szukaj opcji eksportu kodu (Webflow, niektóre AI-buildery). Nie, raczej zostanę na tej samej platformie na lata → mniej istotne kryterium.
- Jaki budżet miesięczny jest dla mnie komfortowy? Wix i Squarespace mają plany zaczynające się od kilkunastu–kilkudziesięciu złotych miesięcznie, Webflow zwykle nieco wyżej, a koszty AI-builderów różnią się mocno między narzędziami — sprawdź aktualny cennik przed wyborem, bo ta kategoria zmienia modele cenowe najczęściej ze wszystkich opisanych w tej książce.

Jeśli po przejściu tej checklisty wciąż wahasz się między dwoma opcjami — wybierz tę, która brzmi jak mniej pracy na start. Żadna z tych decyzji nie jest nieodwracalna, a pierwsza działająca strona jest zawsze lepsza niż idealny plan, który nigdy nie został opublikowany.

Rozdział 5. Sklep internetowy — Shopify

5.1 Shopify czy WooCommerce — jak rozpoznać swoją potrzebę

Jeśli głównym celem Twojej strony jest sprzedaż produktów, masz w praktyce dwie główne drogi, a wybór między nimi warto zrobić świadomie, zanim zaczniesz.

Shopify to platforma w pełni hostowana i zaprojektowana wyłącznie pod e-commerce. Płacisz miesięczny abonament, a w zamian dostajesz koszyk, płatności, magazyn i wysyłkę działające „od ręki”, bez konfigurowania hostingu czy aktualizacji oprogramowania.

WooCommerce to bezpłatna wtyczka, która zamienia zwykłą stronę WordPress (z rozdziału 2) w sklep internetowy. Płacisz tylko za hosting, ale samodzielnie zarządzasz aktualizacjami, bezpieczeństwem i wszystkimi elementami technicznymi.

Wybierz Shopify, jeśli: sprzedaż jest jedynym albo głównym celem strony, chcesz mieć to działające możliwie szybko bez zajmowania się techniczną stroną hostingu, planujesz sprzedawać na wielu kanałach naraz (własna strona, Instagram, Amazon) i jesteś gotów płacić wyższy stały koszt miesięczny w zamian za mniej obowiązków.

Wybierz WooCommerce, jeśli: już masz albo planujesz stronę na WordPressie z innymi treściami (blog, sekcja firmowa) i chcesz dodać do niej sklep, zależy Ci na niższym koszcie długoterminowym, i nie przeszkadza Ci samodzielne zarządzanie aktualizacjami z rozdziału 2.

5.2 Pierwszy sklep — szybki start

Założenie podstawowego sklepu na Shopify zajmuje zwykle od kilku godzin do jednego dnia pracy, w zależności od liczby produktów. Typowa ścieżka:

- Rejestracja na shopify.com i wybór nazwy sklepu (możesz ją zmienić później).
- Wybór motywu — Shopify ma własną bibliotekę darmowych i płatnych motywów dopasowanych pod różne branże (moda, jedzenie, elektronika, rękodzieło).
- Dodanie produktów — dla każdego produktu wypełniasz nazwę, opis, cenę, zdjęcia, a jeśli produkt ma warianty (np. rozmiar, kolor), konfigurujesz je w osobnej sekcji.
- Konfiguracja płatności — najprościej przez Shopify Payments (wbudowane), albo przez zewnętrznego dostawcę płatności, jeśli Shopify Payments nie jest dostępne w Twoim kraju.
- Ustawienie metod i kosztów wysyłki — możesz ustawić stałą cenę, darmową wysyłkę od określonej kwoty zamówienia albo podłączyć kalkulator kurierów.
- Podłączenie domeny i publikacja sklepu.

Checkout (proces finalizacji zamówienia) w Shopify jest jedną z mocniejszych stron platformy — jest dobrze przetestowany i zoptymalizowany pod kątem jak najmniejszej liczby klientów, którzy rezygnują w ostatniej chwili. To realna przewaga w porównaniu z samodzielnie skonfigurowanym checkoutem w WooCommerce.

Podobnie jak WordPress ma wtyczki, Shopify ma własny sklep aplikacji (Shopify App Store) — dodatki do e-mail marketingu, programów lojalnościowych, recenzji produktów i dziesiątek innych funkcji, które instalujesz podobnie łatwo jak wtyczki w WordPressie.

5.3 Pierwszy produkt — dodajemy koszulkę

Zakładam, że masz już konto Shopify po okresie próbnym lub na płatnym planie. Logujesz się do panelu administracyjnego na adresie twojasklep.myshopify.com/admin. Po lewej stronie widzisz menu — klikasz w zakładkę **Produkty**, a następnie w przycisk **Dodaj produkt**.

Ekran dodawania produktu dzieli się na kilka sekcji. Na samej górze wpisujesz **tytuł** — na przykład „Koszulka z nadrukiem — Wilk Północy”. Shopify akceptuje polskie znaki bez żadnych problemów, więc śmiało używaj *ą, ę, ś, ć*. Poniżej tytułu znajduje się pole **opisu**. Tutaj możesz

wstawić zarówno zwykły tekst, jak i formatowanie – pogrubienia, listy – a nawet obrazki i osadzone filmy. Edytor przypomina ten, który znasz z poczty internetowej.

Opis produktu ma znaczenie nie tylko dla kupujących, ale też dla wyszukiwarek. Poświęć mu chwilę – napisz, z jakiego materiału jest koszulka, jaki ma krój, czy nadruk jest sitodrukiem czy DTG, jak prac. Shopify nie ogranicza długości opisu.

W prawej kolumnie ustawiasz **cenę**. Shopify pokazuje trzy pola: cenę porównawczą (dla przekreślonej starej ceny – klasyczny „było 119 zł, teraz 89 zł”), cenę sprzedaży i koszt własny produktu. To ostatnie służy wyłącznie do wyświetlania marży w raportach – nie jest widoczne dla klientów. Wpisujesz 89 zł jako cenę sprzedaży. System akceptuje format 89,00 i 89 – nie musisz się zastanawiać.

Poniżej cen znajduje się sekcja **Warianty**. Klikasz „Dodaj opcje takie jak rozmiar lub kolor”. Shopify proponuje domyślną strukturę: rozmiar i kolor. Zostawiasz obie. W polu rozmiaru wpisujesz wartości oddzielone przecinkiem: S, M, L. W polu koloru: Czarny, Grafitowy, Granatowy. Po zatwierdzeniu Shopify automatycznie generuje dziewięć wariantów – 3 rozmiary × 3 kolory – każdy z własną ceną, kodem SKU i stanem magazynowym.

Dla każdego wariantu możesz ustawić indywidualną cenę – przydatne, gdy większy rozmiar kosztuje więcej – i własne zdjęcie. Klient klikający „Granatowy, L” zobaczy dokładnie tę kombinację, nie zdjęcie główne w innym kolorze. To drobiazg, który realnie zmniejsza liczbę zwrotów.

Przechodzisz do sekcji **Multimedia**. Przeciągasz zdjęcie koszulki z folderu na komputerze. Shopify przyjmuje JPEG, PNG i WebP – nie musisz nic konwertować. Po wgraniu możesz przyciąć zdjęcie i ustawić je jako główne. Dobrą praktyką jest dodanie kilku ujęć: przodu, tyłu, zbliżenia na nadruk, metkę. Zdjęcia w galerii przeciągasz myszą, zmieniając ich kolejność – pierwsze zdjęcie staje się miniaturką widoczną w katalogu sklepu.

Sekcja **Stan magazynowy**. Zaznaczasz „Śledź stany magazynowe” i wpisujesz dostępne sztuki dla każdego wariantu. Shopify może samodzielnie zmniejszać stany po każdym zamówieniu i wysyłać alert, gdy produkt się kończy. Jeśli sprzedajesz produkty cyfrowe albo robisz wszystko na zamówienie – zaznaczasz „Nie śledź”, a produkt będzie zawsze widoczny jako dostępny.

Ostatnia rzecz przed zapisaniem: **Widoczność produktu**. Domyślnie produkt jest niewidoczny, dostępny tylko przez bezpośredni link. Zmieniasz to na „Aktywny” i wybierasz kanały sprzedaży – sklep internetowy, a jeśli masz podłączone – także Facebook, Instagram, Google. Klikasz **Zapisz**.

Cała operacja — od kliknięcia „Dodaj produkt” do zapisania — zajmuje około pięciu minut, gdy robisz to pierwszy raz. Przy piątym produkcie zejdziesz poniżej minuty. Shopify prowadzi Cię przez każdy krok, nie chowa opcji w schowanych zakładkach, a to, co widzisz w panelu, przekłada się bezpośrednio na wygląd strony produktu w sklepie.

5.4 Plany Shopify — który dla kogo

Shopify oferuje trzy plany abonamentowe dla samodzielnych sprzedawców: **Basic**, **Standard** (w panelu często nazywany po prostu „Shopify”) i **Advanced**. Istnieje też opcja **Starter** za niższą cenę, pozwalająca sprzedawać przez linki w social mediach bez osobnej strony — ale do pełnoprawnego sklepu potrzebujesz minimum Basic.

Podstawową różnicą między planami jest **liczba kont pracowników**. W Basic masz do dyspozycji 1–2 konta — wystarczy, gdy prowadzisz sklep samodzielnie. Standard daje do 5 kont, Advanced — do 15. To robi różnicę, gdy zatrudniasz osobę do obsługi klienta, magazyniera albo specjalistę od marketingu i każde z nich potrzebuje własnego loginu z odpowiednimi uprawnieniami.

Drugi czynnik to **raporty**. Basic oferuje podstawowe zestawienia: sprzedaż, zamówienia, produkty. Standard odblokowuje raporty o zachowaniu klientów, segmentacji i marketingu. Advanced daje dostęp do niestandardowych raportów, które konfigurujesz samodzielnie — przydają się, gdy analizujesz marżę na poziomie poszczególnych wariantów albo porównujesz kanały sprzedaży między sobą.

Trzecia warstwa — i dla wielu decydująca — to **prowinzje od transakcji**. Przy korzystaniu z **Shopify Payments** stawki maleją wraz z wyższym planem: Basic ma prowizję wyższą, Standard — średnią, Advanced — najniższą. Dla sklepu, który miesięcznie realizuje kilkaset zamówień, różnica w prowizji między Basic a Advanced potrafi wielokrotnie przekroczyć różnicę w cenie abonamentu. Im więcej sprzedajesz, tym bardziej opłaca się wyższy plan.

Wszystkie trzy plany dają dostęp do tych samych funkcji podstawowych: nieograniczonej liczby produktów, hostingu, certyfikatu SSL, koszyka zakupowego i całodobowego wsparcia technicznego. Shopify nie ogranicza sztucznie liczby zamówień ani ruchu na stronie w zależności od planu — różnica leży wyłącznie w zaawansowanych funkcjach i stawkach prowizji.

Praktyczna strategia: zacznij od **Basic**. To najniższy koszt wejścia, a Shopify umożliwia zmianę planu w dowolnym momencie bez kary i bez utraty danych. Gdy Twój sklep osiągnie

poziom, przy którym oszczędność na prowizjach pokryje różnicę w abonamencie — przejdź na Standard lub Advanced. Nie ma sensu płacić za piętnaście kont pracowników, gdy jedyną osobą obsługującą sklep jesteś Ty.

5.5 Shopify Payments i zewnętrzni dostawcy

Sposób przyjmowania płatności wpływa bezpośrednio na Twój miesięczny koszt prowadzenia sklepu — a wielu początkujących odkrywa to dopiero po pierwszym rozliczeniu.

****Shopify Payments**** to wbudowany system płatności Shopify, dostępny w Polsce od 2024 roku [DO WERYFIKACJI]. Gdy go używasz, Shopify nie dolicza dodatkowej prowizji platformowej od transakcji — płacisz tylko standardową opłatę za przetworzenie płatności kartą. Ta opłata trafia do operatora kart (Visa, Mastercard) i wynosi zwykle niewielki procent kwoty plus drobna stała opłata za każdą transakcję. Konfigurujesz Shopify Payments w kilku kliknięciach, bez podpisywania osobnej umowy z operatorem.

****Zewnętrzni dostawcy płatności**** — Stripe, PayPal, Przelewy24, PayU — to alternatywa. Konfiguracja wymaga założenia konta u operatora, przejścia przez weryfikację i podpięcia go do Shopify przez panel administracyjny. Różnica finansowa jest znacząca: Shopify dolicza własną prowizję od każdej transakcji realizowanej przez zewnętrznego dostawcę. Do tego dochodzi opłata samego operatora. Płacisz więc podwójnie — Shopify za obsługę transakcji na swojej platformie i operatorowi za faktyczne przetworzenie pieniędzy.

W Polsce wygląda to następująco: Shopify Payments obsługuje karty Visa, Mastercard i Maestro — to pokrywa większość płatności kartowych online. Jednak polscy klienci masowo używają BLIKa, który odpowiada za ponad połowę transakcji w krajowym e-commerce. Do tego dochodzą szybkie przelewy online (Przelewy24, PayU) oraz raty i płatności odroczone. Żadnej z tych metod nie obsłużysz przez Shopify Payments — potrzebujesz zewnętrznego dostawcy.

Każdy z tych operatorów ma własną prowizję — zwykle między 1% a 2,5% od kwoty transakcji — do której Shopify doda swoją platformową. Dlatego wiele polskich sklepów na Shopify stosuje rozwiązanie mieszane: Shopify Payments jako główny kanał kartowy, a równolegle Przelewy24 lub PayU jako dodatkową opcję. Klient przy kasie sam wybiera, czy płaci kartą (niższy koszt dla Ciebie), czy BLIK-iem (wyższy koszt, ale popularniejszy wśród kupujących). Shopify pozwala na aktywowanie wielu dostawców jednocześnie.

Konkretny przykład: przy transakcji na 100 zł przez Shopify Payments płacisz orientacyjnie 2 zł prowizji. Ta sama transakcja przez zewnętrznego operatora to 2 zł dla operatora plus dodatkowa opłata Shopify — w zależności od Twojego planu może to być od kilkudziesięciu groszy do kilku złotych więcej. Przy setkach zamówień miesięcznie różnica urasta do poważnej kwoty.

5.6 Koszty ukryte — aplikacje

Gdy pierwszy raz otwierasz Shopify App Store, trudno oprzeć się wrażeniu, że wszystko, czego potrzebujesz, jest na wyciągnięcie ręki. Katalog liczy ponad 13 000 aplikacji, od marketing automation po integracje z hurtowniami. Problem w tym, że wiele z nich działa w modelu abonamentowym, a rachunki sumują się szybciej, niż się spodziewasz.

E-mail marketing to zwykle pierwsza aplikacja, którą instalujesz poza podstawowym zestawem. Shopify ma własne, proste narzędzie do wysyłki maili — wystarczy na start. Gdy potrzebujesz automatyzacji — mail przypominający o porzuconym koszyku, sekwencja powitalna dla nowych subskrybentów — sięgasz po zewnętrzną aplikację. **Mailchimp** oferuje darmowy plan do kilkuset kontaktów. **Klaviyo** — narzędzie stworzone z myślą o e-commerce — też ma darmowy poziom, ale przy większej bazie klientów miesięczny koszt rośnie do kilkudziesięciu złotych.

Recenzje produktów to druga kategoria, która szybko znajduje się na liście „konieczne”. Shopify wycofał własną, darmową aplikację do recenzji. Jeśli chcesz, żeby klienci mogli oceniać produkty i dodawać zdjęcia, instalujesz **Loox** (darmowy plan startowy) albo **Judge.me** (rozbudowany, płatny od określonej liczby zamówień). Sama obecność recenzji zwiększa zaufanie, ale za integrację z mediami społecznościowymi czy możliwość dodawania zdjęć przez klientów często płaci się już osobno.

Kolejne kategorie: **programy lojalnościowe** (naliczanie punktów, nagrody za zakupy), **tłumaczenia sklepu na inne języki**, **narzędzia do dropshippingu** takie jak Oberlo czy Spocket (obie z miesięcznym abonamentem), **czat na żywo** z klientami, **zaawansowane raporty** wykraczające poza to, co daje plan abonamentowy. Lista aplikacji, które „na pewno się przydadzą”, rośnie z każdym tygodniem prowadzenia sklepu.

Doświadczeni sprzedawcy na Shopify mówią wprost: aplikacje potrafią podwoić miesięczny koszt platformy, zanim się zorientujesz. Zasada, która oszczędzi Ci pieniądze na starcie: instaluj

tylko to, bez czego sprzedaż dosłownie nie działa. Podstawowy e-mail marketing masz wbudowany. Recenzje możesz zbierać ręcznie, prosząc klientów o opinię w mailu potransakcyjnym. Shopify Payments pokrywa większość potrzeb płatniczych. Każdą aplikację z miesięcznym abonamentem traktuj jak osobny wydatek, który musi się zwrócić — jeśli aplikacja kosztuje 50 zł miesięcznie, musi generować co najmniej tyle dodatkowej sprzedaży, żeby miała sens.

5.7 Koszty i prowizje — na co uważać

Budżet na Shopify warto liczyć w trzech warstwach, nie tylko po cenie samego abonamentu widocznej na stronie głównej:

- Abonament miesięczny — Shopify ma kilka planów cenowych, rosnących wraz z liczbą dostępnych funkcji (raporty, liczba kont pracowników, niższe prowizje od transakcji). Sprawdź aktualny cennik na shopify.com przed decyzją — zmienia się częściej niż większość rzeczy opisanych w tej książce.
- Prowizje od transakcji — jeśli korzystasz z Shopify Payments, prowizja jest niższa niż przy zewnętrznych dostawcach płatności, do tego doliczana jest standardowa opłata za przetworzenie płatności (zwykle niewielki procent kwoty plus mała stała opłata za transakcję).
- Aplikacje — wiele przydatnych funkcji (rozbudowany e-mail marketing, programy lojalnościowe, zaawansowane raporty) dostępnych jest jako płatne aplikacje z własnym abonamentem, dodawanym do kosztu podstawowego planu.

Dla porównania: WooCommerce nie ma abonamentu za samo oprogramowanie, ale wciąż płacisz hosting (rozdział 6) i prowizję od dostawcy płatności (Stripe, PayPal czy lokalny operator), choć zwykle nie ponosisz dodatkowej prowizji „platformowej” tak jak przy płatnościach niewbudowanych w Shopify.

Praktyczna rada na start: zanim zarejestrujesz pełny płatny plan, skorzystaj z darmowego okresu próbnego Shopify, żeby skonfigurować sklep i sprawdzić, czy platforma odpowiada Twoim potrzebom, zanim zaczniesz płacić.

Rozdział 6. Hosting i domena

6.1 Jak wybrać i kupić domenę

Domena to adres Twojej strony — to, co odwiedzający wpisuje w przeglądarce, np. twojafirma.pl. Niezależnie od tego, którą drogę z poprzednich rozdziałów wybrałeś, w pewnym momencie będziesz musiał ją kupić.

Kilka praktycznych zasad przy wyborze nazwy:

- Krótka i łatwa do zapamiętania — jeśli musisz ją komuś dyktować przez telefon, długa nazwa z trudnymi do przeliterowania słowami szybko staje się problemem.
- Unikaj myślników i cyfr, jeśli to możliwe — łatwo o nich zapomnieć albo źle je przekazać ustnie.
- Dopasuj rozszerzenie do odbiorcy — .pl dobrze sygnalizuje lokalny, polski biznes; .com jest uniwersalnym standardem, jeśli myślisz o odbiorcach również poza Polską.
- Sprawdź dostępność na kilku rozszerzeniach naraz — narzędzia rejestratorów domen pokazują od razu, czy ta sama nazwa jest wolna jako .pl, .com, .eu itd.

Domeny kupuje się u rejestratorów — firm takich jak OVH, home.pl, nazwa.pl czy Namecheap, często też w ramach pakietu razem z hostingiem (rozdział 6.2). Ważne, by zrozumieć: domeny nie kupujesz raz na zawsze, tylko wynajmujesz na rok (lub kilka lat z góry) — jeśli zapomnisz o przedłużeniu, ktoś inny może ją zarejestrować po wygaśnięciu. Włącz automatyczne odnawianie od razu przy zakupie, żeby nie musieć o tym pamiętać.

Warto też sprawdzić, czy rejestrator oferuje ochronę prywatności danych właściciela domeny (czasem darmową, czasem za dodatkową opłatą) — bez niej Twoje dane kontaktowe podane przy rejestracji mogą być publicznie widoczne w bazie WHOIS.

6.2 Rodzaje hostingu

Hosting to „miejscę”, gdzie fizycznie znajdują się pliki Twojej strony — serwer, który odpowiada na zapytania przeglądarek odwiedzających. Dla pierwszej strony realnie liczą się trzy typy:

Hosting współdzielony (shared) — najtańsza opcja, w której Twoja strona dzieli zasoby serwera z wieloma innymi stronami. Wystarczający dla większości wizytówek i mniejszych blogów na początku. To, od czego warto zacząć, jeśli nie wiesz jeszcze, ile ruchu będzie miała Twoja strona.

Hosting zarządzany dla WordPressa — droższy wariant hostingu współdzielonego, zoptymalizowany specjalnie pod WordPressa: automatyczne aktualizacje, wbudowana pamięć podręczna (cache) i dodatkowe zabezpieczenia skonfigurowane od razu, bez Twojej ingerencji. Jeśli wybrałeś drogę z rozdziału 2 i wolisz mniej martwić się o szczegóły techniczne z podrozdziałów 2.11–2.12, to dobra inwestycja.

VPS (Virtual Private Server) — wydzielony kawałek mocniejszego serwera, z większą kontrolą i zasobami, ale wymagający więcej wiedzy technicznej do samodzielnej konfiguracji. To opcja na później, kiedy Twoja strona wyrośnie z hostingu współdzielonego — nie na pierwszy dzień.

Dla pierwszej strony z tej książki: hosting współdzielony (lub zarządzany dla WordPressa, jeśli wybrałeś rozdział 2) w 95% przypadków wystarczy w zupełności. Buildery i AI-narzędzia z rozdziału 4 oraz Shopify z rozdziału 5 mają hosting wbudowany w cenę abonamentu — ten podrozdział dotyczy głównie ścieżki WordPress i HTML/CSS.

6.3 SSL, e-mail na domenie, podstawy DNS

Trzy techniczne elementy, o których warto wiedzieć, nawet nie wgłębiając się w szczegóły:

SSL (certyfikat bezpieczeństwa) szyfruje połączenie między przeglądarką odwiedzającego a Twoim serwerem — to ta sama różnica, która zmienia adres strony z http:// na https:// i pokazuje kłódkę przy adresie w przeglądarce. W 2026 roku to absolutny standard, bez którego przeglądarki ostrzegają odwiedzających, że strona jest „niezabezpieczona”, a Google traktuje to jako sygnał

obniżający pozycję w wynikach wyszukiwania. Dobra wiadomość: prawie każdy hosting w 2026 roku instaluje darmowy certyfikat SSL (zwykle przez Let's Encrypt) automatycznie, bez dodatkowej konfiguracji z Twojej strony.

E-mail na własnej domenie (np. kontakt@twojafirma.pl zamiast twojafirma@gmail.com) wygląda dużo bardziej profesjonalnie i buduje zaufanie. Większość pakietów hostingowych pozwala stworzyć taką skrzynkę w panelu w kilka minut, choć dostępne miejsce na wiadomości bywa ograniczone — przy większych potrzebach pocztowych warto rozważyć osobną usługę (np. Google Workspace) podłączoną do tej samej domeny.

DNS (System Nazw Domen) to mechanizm, który łączy Twoją domenę (nazwę) z faktycznym serwerem, na którym stoi strona (adresem technicznym). Kiedy kupujesz hosting osobno od domeny, musisz „wskazać” w panelu domeny, gdzie ma ona prowadzić — zwykle przez wpisanie adresów serwerów nazw (nameserverów) podanych przez firmę hostingową. Zmiany w DNS nie są natychmiastowe — mogą rozprzestrzeniać się po internecie od kilkunastu minut do 24–48 godzin, więc nie panikuj, jeśli strona nie działa „od razu” po zmianie ustawień.

6.4 GitHub Pages i inne darmowe opcje — kiedy mają sens

Jeśli zbudowałeś swoją stronę zgodnie z projektem z podrozdziału 3.7 — czysty HTML i CSS, bez bazy danych — masz dostęp do kategorii hostingu, która dla pozostałych dróg z tej książki po prostu nie istnieje: darmowy hosting stron statycznych.

GitHub Pages to bezpłatna usługa, która publikuje Twoją stronę prosto z repozytorium kodu na GitHubie. W praktyce: wgrywasz pliki index.html i styl.css do repozytorium, włączasz GitHub Pages w ustawieniach, i Twoja strona jest dostępna pod darmowym adresem w ciągu kilku minut — z możliwością podłączenia własnej domeny później.

Podobne darmowe (na start) opcje to Netlify i Vercel — odrobinę bardziej rozbudowane, z automatycznym wdrażaniem zmian od razu po aktualizacji plików, popularne też jako miejsce hostowania kodu wyeksportowanego z niektórych AI-builderów z rozdziału 4.

Kiedy ma to sens: dla projektów hobbystycznych, portfolio, prostych stron osobistych, stron technicznych dokumentacji — wszystkiego, co nie wymaga bazy danych ani logowania użytkowników. Kiedy nie ma sensu: dla WordPressa (wymaga PHP i bazy danych, których te usługi nie obsługują) czy sklepu internetowego z dynamicznym koszykiem zakupów. To realna, w pełni profesjonalna opcja dla odpowiedniego przypadku — ale dotyczy wyłącznie ścieżki HTML/CSS z rozdziału 3, nie pozostałych dróg z tej książki.

Rozdział 7. Co wybrać — podsumowanie

7.1 Drzewko decyzyjne: od pytania do rekomendacji

Jeśli przeczytałeś tę książkę od początku, prawdopodobnie już wiesz, w którą stronę idziesz. Ten podrozdział to skrócona ścieżka dla osób, które chcą szybko zweryfikować swój wybór albo wracają do książki po dłuższej przerwie.

Pytanie 1: Czy głównym celem strony jest sprzedaż produktów?

→ Tak: przejdź do pytania 1a.

→ Nie: przejdź do pytania 2.

Pytanie 1a: Czy oprócz sklepu chcesz mieć też bloga albo rozbudowaną treść firmową?

→ Tak, sklep i treść razem → WooCommerce na WordPressie (rozdziały 2 i 5).

→ Nie, czysto sklep, bez dodatkowej treści → Shopify (rozdział 5).

Pytanie 2: Ile czasu chcesz poświęcić, zanim strona będzie gotowa?

→ Minuty do godzin, jak najmniej wysiłku → przejdź do pytania 2a.

→ Mogę poświęcić kilka weekendów w zamian za większą kontrolę → przejdź do pytania 3.

Pytanie 2a: Łatwiej Ci opisać stronę słowami, czy poukładać ją wizualnie, klikając?

→ Opisać słowami → AI-builder (rozdział 4.5).

→ Poukładać wizualnie → Wix lub Squarespace (rozdziały 4.2–4.3).

Pytanie 3: Czy zależy Ci na pełnej niezależności (własny hosting, bez stałego abonamentu za platformę) i długoterminowym rozwoju strony?

→ Tak → WordPress (rozdział 2).

→ Niezupełnie, ale chcę dużej kontroli wizualnej bez pisania kodu → Webflow (rozdział 4.4).

→ Chcę naprawdę zrozumieć, jak to działa, niezależnie od czasu → HTML/CSS od podstaw (rozdział 3).

Żadna z tych odpowiedzi nie jest ostateczna do końca życia Twojej strony – wiele osób zaczyna od no-code albo AI-buildera, żeby szybko ruszyć z miejsca, a po roku czy dwóch, kiedy strona i potrzeby rosną, przechodzi na WordPressa po większą kontrolę. To normalna droga, nie porażka pierwszego wyboru.

7.2 Tabela porównawcza wszystkich dróg

	WordPress	HTML/CSS	No-code (Wix/ SquareSpace)	AI-buildery	Shopify
Czas na start	Dni	Tygodnie (nauka)	Godziny	Minuty– godziny	Godziny– dzień
Koszt początkowy	Niski (hosting+domena)	Bardzo niski	Niski–średni	Niski–średni	Średni
Koszt długoterminowy	Niski, bez abonamentu za platformę	Bardzo niski	Stały abonament	Stały abonament	Stały abonament + prowizje
Kontrola i elastyczność	Bardzo wysoka	Najwyższa	Średnia	Średnia, zależna od narzędzia	Wysoka w obrębie e- commerce
Krzywa uczenia	Średnia	Wysoka	Bardzo niska	Bardzo niska	Niska– średnia
Ryzyko vendor lock-in	Niskie	Brak	Wysokie	Zmienne (sprawdź eksport)	Średnie
Najlepsze dla	Stron rozwijających się latami	Nauki i pełnej kontroli	Szybkiej, ładnej wizytówki	Najszybszego pierwszego efektu	Sklepów jako głównego celu

Tabela nie zastępuje rozdziałów, z których pochodzi – to ściągawka do wykorzystania po przeczytaniu, nie zamiast.

7.3 Co dalej po pierwszej stronie

Niezależnie od wybranej drogi, kilka rad dotyczy każdej z nich w równym stopniu.

Opublikowana strona jest lepsza niż idealny plan. Łatwo wpaść w pułapkę dopracowywania każdego szczegółu przed publikacją. Lepiej opublikować wersję, z której jesteś w 80% zadowolony, i poprawiać ją na bieżąco, niż czekać na 100%, które nigdy nie nadejdzie.

Zainstaluj podstawową analitykę od pierwszego dnia — np. Google Analytics i Google Search Console (darmowe, działają z każdą drogą opisaną w tej książce). Dzięki nim po kilku tygodniach zobaczysz prawdziwe dane: skąd przychodzą odwiedzający, które podstrony czytają, gdzie rezygnują. To znacznie lepszy przewodnik do kolejnych zmian niż własne przeczucia.

Wróć do innych rozdziałów, kiedy zmienią się Twoje potrzeby. Strona, którą postawiłeś w Wixie jako wizytówka, może za rok potrzebować bloga i lepszego SEO — wtedy warto wrócić do rozdziału 2. Sklep stawiany od razu w WooCommerce może z czasem przerosnąć Twoje możliwości techniczne — wtedy warto wrócić do rozdziału 5 i Shopify. Żadna z tych dróg nie jest decyzją na zawsze.

I jedna rzecz na koniec: ta książka opisuje stan rynku w połowie 2026 roku. Konkretnie narzędzia, ceny i nazwy platform będą się zmieniać — to, co prawdopodobnie się nie zmieni, to pytania z podrozdziału 7.1. Jeśli za kilka lat wrócisz do tej książki przy kolejnej stronie, te pytania wciąż będą dobrym punktem startu, nawet jeśli odpowiedzi po prawej stronie wymagają już aktualizacji.

Powodzenia z Twoją pierwszą stroną.

Słowniczek pojęć

AI-builder — narzędzie, w którym opisujesz słowami, jak ma wyglądać Twoja strona, a sztuczna inteligencja generuje ją za Ciebie (np. Lovable, Wix AI, Framer AI). Rozdział 4.5.

Backup (kopia zapasowa) — zapisana wersja całej strony, do której możesz wrócić, jeśli coś pójdzie nie tak. Rozdział 2.11.

Box model — sposób, w jaki przeglądarka traktuje każdy element strony jako prostokątne „pudełko” złożone z treści, paddingu, obramowania i marginesu. Rozdział 3.4.

Cache (pamięć podręczna) — mechanizm przyspieszający wczytywanie strony poprzez tymczasowe zapisywanie jej elementów, żeby nie generować ich od nowa przy każdej wizycie. Rozdział 2.9.

Checkout — proces finalizacji zamówienia w sklepie internetowym, od koszyka do potwierdzenia płatności. Rozdział 5.2.

CMS (Content Management System) — system do zarządzania treścią strony przez panel administracyjny, bez pisania kodu od zera. WordPress jest najpopularniejszym przykładem. Rozdział 2.1.

CSS (Cascading Style Sheets) — język opisujący wygląd strony: kolory, czcionki, odstępy, układ elementów. Rozdział 3.3.

Domena — adres Twojej strony w internecie, np. twojafirma.pl. Wynajmowana rok po roku, nie kupowana raz na zawsze. Rozdział 6.1.

DNS (System Nazw Domen) — mechanizm łączący nazwę domeny z faktycznym serwerem, na którym znajduje się strona. Rozdział 6.3.

DOM — drzewiasta struktura, którą przeglądarka tworzy z kodu HTML strony, zanim nałoży na nią styl z CSS. Rozdział 3.1.

Flexbox — narzędzie w CSS do układania elementów względem siebie, np. w rzędku lub kolumnie. Rozdział 3.4.

GitHub Pages — darmowa usługa hostowania prostych, statycznych stron (HTML/CSS) prosto z repozytorium kodu. Rozdział 6.4.

Gutenberg — domyślny edytor treści w WordPressie, w którym budujesz strony i wpisy z bloków. Rozdział 2.6.

Hosting — miejsce, gdzie fizycznie znajdują się pliki Twojej strony i które odpowiada na zapytania przeglądarek odwiedzających. Rozdział 6.2.

HTML (HyperText Markup Language) — język opisujący strukturę i treść strony za pomocą znaczników. Rozdział 3.2.

Media query — reguła w CSS, która stosuje określone style tylko przy danej szerokości ekranu — podstawa responsywności. Rozdział 3.5.

Motyw (theme) — szata graficzna strony w WordPressie lub innym builderze: kolory, układ, typografia. Rozdział 2.5.

No-code — narzędzia do budowy stron bez pisania kodu, działające na zasadzie wizualnego przeciągania elementów (np. Wix, Squarespace, Webflow). Rozdział 4.1.

Permalink — struktura adresu URL danej podstrony lub wpisu w WordPressie. Rozdział 2.4.

Responsywność — zdolność strony do dostosowywania swojego wyglądu do różnych rozmiarów ekranu, zwłaszcza telefonu. Rozdział 3.5.

SEO (Search Engine Optimization) — działania zwiększające widoczność strony w wynikach wyszukiwania Google. Rozdział 2.10.

SSL / HTTPS — szyfrowanie połączenia między przeglądarką a serwerem, widoczne jako kłódka przy adresie strony. Rozdział 6.3.

VPS (Virtual Private Server) — wydzielony, mocniejszy kawałek serwera z większą kontrolą, odpowiedni dla rozwiniętych stron. Rozdział 6.2.

Vendor lock-in — sytuacja, w której trudno przenieść stronę do innego dostawcy, bo platforma nie pozwala na pełny eksport danych. Rozdział 4.6.

Viewport — widoczny obszar strony na ekranie urządzenia; ustawienie viewportu w kodzie decyduje, jak strona zachowuje się na telefonie. Rozdział 3.5.

WooCommerce — darmowa wtyczka zamieniająca stronę WordPress w sklep internetowy. Rozdział 5.1.

WordPress.com vs WordPress.org — WordPress.com to hostowana usługa z ograniczeniami; WordPress.org to darmowe oprogramowanie instalowane na własnym hostingu, z pełną kontrolą. Rozdział 2.2.

Wtyczka (plugin) — dodatek rozszerzający funkcje WordPressa, np. formularz kontaktowy, zabezpieczenia czy SEO. Rozdział 2.9.

Koniec książki

Moja pierwsza strona internetowa







