ChampSoft
The Software Visionaries

Issue No 50 - October 1st 2026

NEWSLETTER

Volume 50



FEATURED Page 01 - 02

AI Won't Replace Developers

But Developers Who Use AI Will Replace Those Who Don't

Web Dev & Security

How Passkeys Are Changing Website Authentication

For decades, passwords have been the default way to protect online accounts. Yet they create a familiar list of problems: people forget them, reuse them across services, choose weak combinations, and can be tricked into entering them on fraudulent websites.

Page 03 - 04

Productivity & Wellbeing

Small Habits, Big Difference

We spend a big part of our week at work, so how we feel during those hours matters. The good news is that looking after your wellbeing doesn't need a big lifestyle change. Small, consistent habits do most of the work. Here are a few you can start today.

Page 05 - 06

champsoft.com — Blogs Live

Latest Blogs

ERP & Supply Chain + Autonomous Coding Agents + Custom Software Costs + Medical Invoicing + Mirth Connect 4.7.1 + Outsourced Software ROI

AI Won't Replace Developers

But Developers Who Use AI Will Replace Those Who Don't

4 minute read

A few years ago, being a software developer meant spending hours writing code, searching documentation, debugging errors, and occasionally arguing with a compiler that was technically always right. Today, things are different.

You can describe what you want in plain English, and AI can generate a working function in seconds. It can explain unfamiliar code, write tests, suggest fixes, create documentation, and even help design an application.

Naturally, one question keeps coming up:

> **"Will AI replace software developers?"**

I don't think that's the right question. The better question is:

> **"What happens to developers who refuse to use AI?"**

AI Is Changing the Job, Not Eliminating It

AI is incredibly good at producing code. But software development has never been only about writing code. A real developer has to understand the problem before solving it. Imagine a product manager says, "We need a login system."

An AI can quickly generate authentication code. But should the application use passwords, OAuth, passkeys, or single sign-on? How should sessions be managed? What happens when an account is compromised? What data should be stored? How should permissions work? These aren't simply coding questions. They're engineering decisions.

The Biggest Advantage Isn't Writing Code Faster

The obvious benefit of AI is speed. A developer who previously spent an hour creating boilerplate code might now do it in ten minutes.

But the bigger advantage is not saving ten minutes. It's what you do with the extra time. Instead of spending most of your day writing repetitive code, you can spend more time thinking about architecture, performance, security, user experience, and edge cases. **AI can handle some of the mechanical work, while developers focus more on the thinking work.** That's a significant shift.

But There's a Catch

AI-generated code can look convincing. And that's exactly what makes it dangerous. AI can confidently produce code that is outdated, inefficient, insecure, or simply wrong.

It might use an API that doesn't exist. It might introduce a subtle security vulnerability. It might solve the immediate problem while creating a much bigger maintenance problem six months later.

If you don't understand the code AI generates, you're not really using AI as a developer. You're gambling.

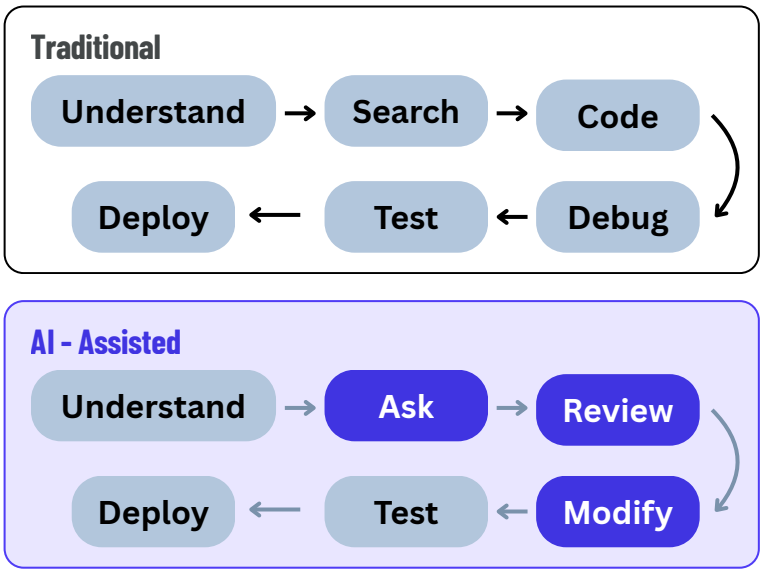
> **The most valuable skill in the AI era isn't knowing how to generate code.**
>
> **It's knowing whether the generated code is worth keeping.**

AI can help developers make those decisions, but someone still needs to understand the business requirements, evaluate the risks, question assumptions, and take responsibility for the final result.

That person is the developer.

The New Developer Workflow

The traditional workflow looked something like this, compared with what an AI-assisted one might look like:



Notice something important. The developer hasn't disappeared. The developer has moved up the abstraction level.

Instead of manually writing every line, developers increasingly describe problems, evaluate solutions, and guide AI toward the desired result. In other words, the developer becomes less like a typist and more like an **engineer directing another extremely fast junior developer.**

And just like you wouldn't blindly merge code written by a junior developer, you shouldn't blindly merge code written by AI.

AI Won't Make Everyone a Great Developer

There is another uncomfortable truth. AI doesn't eliminate the difference between a beginner and an experienced developer. In some situations, it makes that difference even more obvious.

An experienced developer can look at AI-generated code and quickly recognize:

- "This won't scale."
- "This creates a security problem."
- "There's a race condition here."
- "We don't need this abstraction."

A beginner may simply think:

> **"The code works. Ship it."**

That's where experience becomes incredibly valuable. The ability to judge code may become more important than the ability to produce it.

The Developers Who Adapt Will Win

Technology has always changed the tools developers use. We moved from machine code to assembly, from assembly to high-level languages, from local servers to cloud platforms, and now from manually written code toward AI-assisted development.

Developers didn't disappear. The tools changed. AI is simply the next major shift. So instead of being afraid that AI will take your job, learn how to make AI better at your job.

- + Explore unfamiliar technologies
- + Generate test cases
- + Review your code
- + Explain complicated systems
- + Automate repetitive tasks

But most importantly, keep your engineering judgment sharp. Because the future probably won't belong to developers who write the most code. It will belong to developers who can solve the right problems, make good decisions, and use AI to move faster.

AI isn't coming for developers. It's coming for the parts of development that don't require a developer's full attention. And that's actually good news.

The future of software development may not be humans versus AI.

It may simply be developers with AI versus developers without it.



RUSHIKESH CHAUDHARI
Software Engineer
ChampSoft
The Software Visionaries

WEB DEVELOPMENT • SECURITY • 2026

Beyond Passwords: How Passkeys Are Changing Website Authentication

A simpler, phishing-resistant approach to signing in is moving into the mainstream.

Why it matters: Passkeys replace shared passwords with cryptographic credentials that are unlocked using the same biometric, PIN, or device authentication people already use.

01

Passwords Have Reached Their Limits

For decades, passwords have been the default way to protect online accounts. Yet they create a familiar list of problems: people forget them, reuse them across services, choose weak combinations, and can be tricked into entering them on fraudulent websites.

Adding one-time codes or approval prompts improves security, but many of these methods can still be targeted by phishing and also add friction to the sign-in experience.

02

What Exactly Is a Passkey?

A passkey is a password-replacement credential based on public-key cryptography. When a user creates one, the website keeps a public key while the corresponding private credential remains protected by the user's device or passkey provider. To sign in, the user typically confirms their identity with a fingerprint, face recognition, device PIN, or another local unlock method. The biometric information itself stays on the device; it is not sent to the website.

Select account → **Approve sign-in** fingerprint, face or PIN → **Continue**

For users, the experience can feel much simpler: select an account, approve the sign-in, and continue. There is no password to remember or type, and passkeys can be securely synchronized across supported devices or stored as device-bound credentials.



03

What Exactly Is a Passkey?

The biggest advantage is phishing resistance. A passkey is tied cryptographically to the legitimate website or service, so a credential created for one site cannot simply be entered into a look-alike phishing page. Passkeys also remove the shared password secret that attackers often seek through credential stuffing or stolen password databases. This makes authentication both harder to steal and easier for legitimate users.

04

What This Means for Web Developers

For developers, adopting passkeys is more than adding a new login button. Websites generally implement them through WebAuthn, the browser standard behind passkey authentication. In August 2026, WebAuthn Level 3 became a W3C Recommendation, providing a stable modern specification for browser-based public-key authentication.

A successful implementation also needs thoughtful user experience design. Teams must consider passkey enrollment, sign-in prompts, multiple devices, account recovery, legacy authentication, accessibility, and what happens when a user loses or replaces a device. During migration, many services will continue offering passwords alongside passkeys while users become comfortable with the newer method.



05

A Shift Already Underway

Passkeys are no longer an experimental idea. FIDO Alliance research published in 2026 reported five billion passkeys in active use and found that 75% of surveyed consumers had enabled passkeys on at least some accounts. The same research found that 68% of surveyed organizations were deploying, piloting, or rolling them out for workforce authentication.

5B
passkeys in active use

75%
of surveyed consumers have enabled passkeys on at least some accounts

68%
of surveyed organizations are deploying, piloting, or rolling out passkeys for workforce authentication

Those figures point to a broader change in how the web approaches identity: security and convenience no longer have to be opposing goals.

06

The Future of Sign-In

Passwords will not disappear overnight. Existing systems, recovery requirements, user habits, and organizational policies will keep them around for some time. But the direction is becoming clearer.

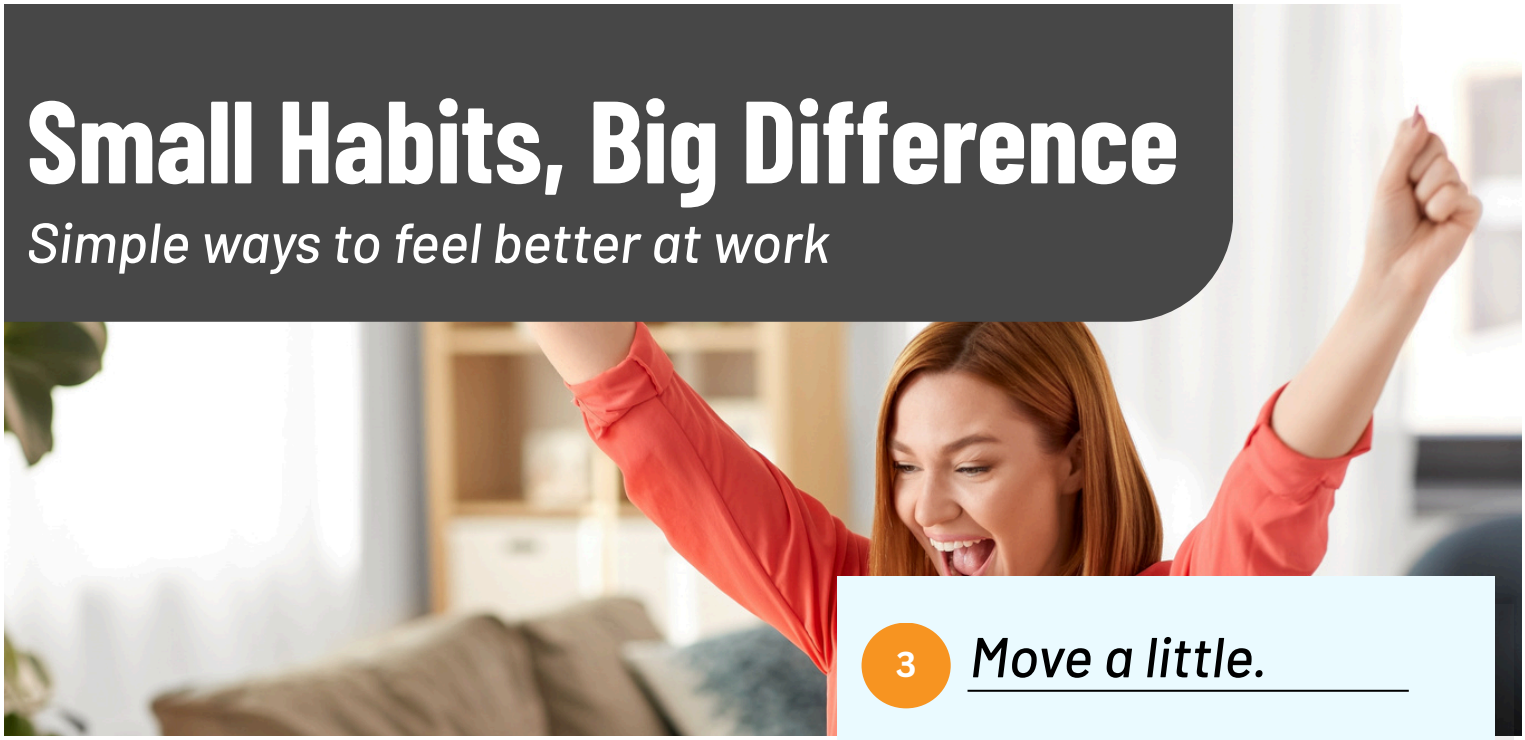
As browser and platform support matures, passkeys give web teams an opportunity to design authentication around cryptographic proof instead of something users must remember. For developers and businesses alike, that can mean fewer password-related frustrations, stronger protection against phishing, and a smoother path into the next generation of website security.



RAVISHANKAR NIRANJAN
Web Developer
Champ360
MARKETING

Small Habits, Big Difference

Simple ways to feel better at work



We spend a big part of our week at work, so how we feel during those hours matters. The good news is that looking after your wellbeing doesn't need a big lifestyle change. Small, consistent habits do most of the work. Here are a few you can start today.

1 Take real breaks

Stepping away from your screen for five minutes every hour or so helps you reset. Stretch, walk to the window, or refill your water bottle. Eating lunch away from your desk, even once or twice a week, makes a noticeable difference to your afternoon energy.

2 Protect your focus.

Constant notifications make the day feel busier than it is. Try setting aside one or two blocks of time for focused work, muting non-urgent alerts, and checking messages at set times instead of all day.

3 Move a little.

You don't need a gym session. Take the stairs, hold a walking meeting, or do a few desk stretches. Even light movement helps with posture, mood, and concentration.



4 Set boundaries after hours.

Switching off matters. Decide when your day ends, and try not to check emails late in the evening. Rest keeps you sharp the next day.

5 Talk to someone.

Feeling overwhelmed is normal, and it's nothing to hide. A quick chat with a colleague, your manager, or HR can lighten the load. Asking for help early shows good judgment, not weakness.

5 Celebrate small wins.

Finished a tough task? Helped a teammate? Take a moment to notice it. Appreciation, whether given or received, builds a positive and supportive workplace

6 Celebrate small wins.

Finished a tough task? Helped a teammate? Take a moment to notice it. Appreciation, whether given or received, builds a positive and supportive workplace

A CLOSING THOUGHT

Wellbeing isn't a one-time event. It's built through everyday choices, and a healthy team does better work and supports each other better. This month, pick just one habit from this list and try it for two weeks. Small steps add up.



SAZIA BEGUM
HR - Executive
ChampSoft
The Software Visionaries



Welcome To CHAMP!

We're thrilled to welcome our newest members to the ChampSoft family. Wishing you all the best as you begin this exciting journey with us together, let's achieve great things!



RUSHAN DE SILVA
ASSOCIATE PROJECT MANAGER
ChampSoft
The Software Visionaries



IRAJ ANUSHKA
TRAINEE SOFTWARE ENGINEER
ChampSoft
The Software Visionaries



DINITHI DINUSHIKA
TRAINEE DATA ENGINEER
ChampSoft
The Software Visionaries



THARUSHA SEWMINI
TRAINEE SOFTWARE ENGINEER
ChampSoft
The Software Visionaries



YUWANI VIHANGA
TRAINEE MACHINE LEARNING ENGINEER
DQI
DEEP DATA INSIGHT



RAVISHANKAR NIRANJAN
WEB DEVELOPER
Champ360
MARKETING

CATCH OUR LATEST BLOGS

Read the Latest on - champsoft Blogs



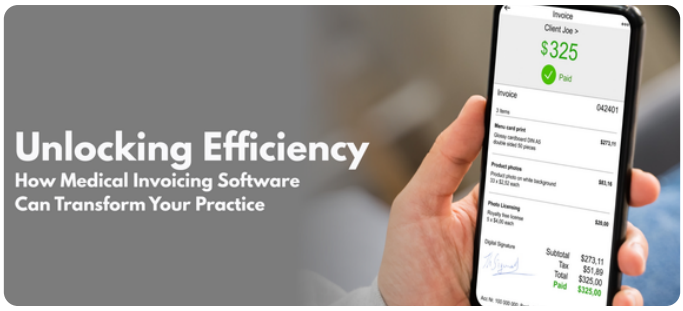
Why Developers Are Losing Control of Autonomous Coding Agents And How to Get It Back

Developers lose control over autonomous coding agents when agents commit code, install dependencies, or call external tools without required approval gates, audit trails, or scoped permissions



Why Custom Software Is Cheaper Than You Think

If you've ever written off custom software as "too expensive," you're not alone, and you're probably wrong. Most business owners assume off-the-shelf tools are the budget-friendly option. But when you add up licensing fees, workaround costs,



Unlocking Efficiency: How Medical Invoicing Software Can Transform Your Practice

Medical invoicing software automates charge capture, coding, claims submission, and patient statements, connecting billing directly to cloud EHR and telehealth data. Practices



Mirth Connect 4.7.1 Upgrade Guide: Java 17 Migration, Compatibility Testing, and Zero-Downtime Deployment

Mirth Connect 4.7.1 requires Java 17 as its minimum runtime, following the switch introduced in version 4.7.0. Upgrading safely means auditing ,



Maximizing ROI: How to Effectively Manage Outsourced Software Product Development Projects

Maximizing ROI on outsourced software product development means governing the engagement, not just picking a vendor: define KPIs and success



How Enterprise Resource Planning and Supply Chain Management Drive Business Success: Insights from Industry Leaders

Enterprise resource planning and supply chain management fuel business success by unifying inventory, procurement, and supplier data into one real-time system.