

# Offshore, Nearshore or Onshore Java development: Solving the age-old question

*Java development teams have a choice between offshore, nearshore, and onshore when deciding where to locate developers. Each model has distinct pros and cons to consider regarding communication, collaboration, cost, access to talent, and more.*

Temps de lecture : minute

---

19 September 2024

This article explores the key factors technology leaders should weigh when determining the right approach for their Java projects and organization.

## Defining key terms

Before analysing each model, it helps to define offshore, nearshore and onshore precisely:

### Offshore Java Development

Offshore Java teams are located in a country far from the client base, often overseas. Popular *offshore Java development* destinations include India, Eastern Europe and Latin America.

### Nearshore Java Development

Nearshore Java developers reside in closer proximity to the client, but still in a separate country. For U.S.-based companies, common nearshore

locations include Canada, Mexico and Central America.

## Onshore Java Development

Onshore Java teams work in the same country as the business they serve. For a company headquartered in the U.S., its onshore developers also live in the U.S.

Now, let's explore the advantages and challenges unique to each model.

## Key considerations: Offshore Java Development

Establishing an offshore Java team attracts organisations with the promise of significant cost savings. However, several factors can minimize those gains or even eliminate them entirely.

### Offshore development cost savings

The cost of living in popular outsourcing regions pales in comparison to Western countries. Thus, the salaries software developers command reflect that. This means companies can build large teams with top-tier talent for a fraction of the price compared to onshore developers.

However, hidden costs can creep up over time:

- Travel expenses for occasional in-person collaboration
- Productivity/delivery lags due to communication barriers
- Turnover when developers leave for better-paying jobs
- Building adequate internal infrastructure/tools to enable collaboration

These factors can diminish the return on investment of an offshore team.

### Communication challenges

Managing distributed software teams thousands of miles away in a drastically different time zone is inherently challenging. Language and cultural barriers further complicate communication.

Without careful orchestration, teams easily lose productivity from:

- Misaligned priorities/requirements
- Delayed feedback due to time zone gaps
- A lack of shared context and personal connections

These issues apply to nearshore arrangements as well, but generally not to the same extent.

Limited access to specialised skill sets

Ukraine has a deep pool of Java talent that is perfect for bread-and-butter application development. However, organisations may struggle to find offshore team members possessing specialised skill sets like:

- Expertise in emerging frameworks/tools
- Niche domain experience
- Background in newer languages like Kotlin

Since Java emerged in the mid-90s, it's become less ubiquitous in newer tech epicentres like Eastern Europe. Companies seeking leading-edge capabilities may have better luck engaging onshore or nearshore developers.

Cultural differences

Navigating cultural differences requires patience and nuance. For example, some cultures tend to avoid saying "no" directly. Leaders who don't understand this may misinterpret vague assurances as concrete confirmation. Other cultures have different attitudes about timeliness,

workplace hierarchy and more.

Without cultural training, teams risk problems like:

- Misset expectations around project commitments
- Communication breakdowns
- Low morale/engagement

Information security concerns

Storing sensitive customer data with offshore vendors increases privacy and security risks. If proprietary source code and other intellectual property get leaked, the financial implications could be severe.

Rigorous data protection measures are essential for mitigating these threats. Disciplined protocols must be established, communicated and enforced at all times.

For some organisations, the information security threat poses too much liability. In these cases, onshore or nearshore alternatives may provide more prudent options.

## Key considerations: Nearshore Java Development

The nearshore development approach attempts to strike a balance between offshore cost savings and onshore alignment. For U.S. companies, nearshore destinations in Latin America offer a compromise:

- Physical proximity for simpler travel/collaboration
- Cultural/language familiarity
- Lower rates than onshore developers

These factors fuel many benefits—along with a unique set of

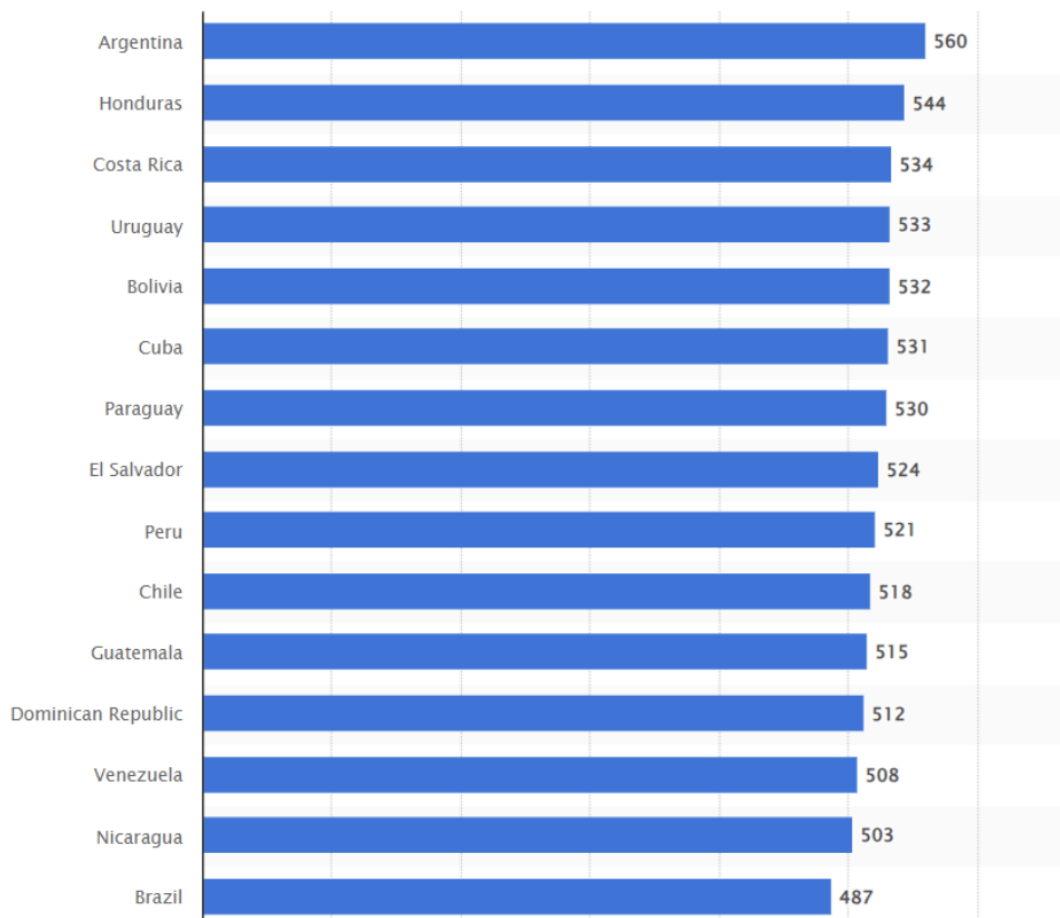
considerations.

## Communication ease

Compared to offshore teams halfway around the world, nearshore developers operate in similar or identical time zones. This allows for high-bandwidth communication during normal business hours:

- Quick questions/answers
- Low latency on video calls
- Rapid feedback loops

Shared language and cultural understanding also smooth coordination. According to *Statista*, many nearshore partners speak English at a decent level.



## Talent pool alignment

Nearshore regions offer strong talent pools of Java professionals:

- Mexico has 225,000 engineers
- Colombia's talent pool is 62,000+ IT professionals
- Costa Rica has approximately 22,000 developers

Salaries for developers in these regions fall below U.S. ranges as well.

This allows companies to build robust teams with advanced capabilities more economically than with onshore staffing.

Travel convenience

Short 2-4 hour flights make periodic in-person collaboration with nearshore teams feasible. By contrast, 22-hour treks to India or China pose bigger barriers to firsthand interaction.

The relative ease of site visits and executive meetings in nearshore locations facilitates stronger partnerships.

Cultural synergy

Shared time zones aren't the only cultural alignment nearshore arrangements provide. Similar cultural attitudes, values and communication styles also promote harmony:

- Strong work ethic
- Focus on relationship-building
- Low power distance index (PDI)

Low PDI cultures, like those in Latin America, emphasise flat organisational structures and accessibility between executives and staff. This facilitates transparency and agility.

Of course, companies should still invest in cultural training to prevent

potential misunderstandings. But nearshore alignment minimizes these risks.

### Consistent business context

Nearshore countries operate in closer proximity not just geographically but economically as well. Since they share time zones with their clients, nearshore teams directly experience the same business context.

For example, partners in Colombia instantly recognise how an unexpected regulatory change in the U.S. could impact operations. This real-time understanding keeps priorities aligned even amidst sudden market shifts.

No doubt nearshoring has its benefits. But it's not necessarily the right fit for every organisation as we'll explore next.

## Key considerations: Onshore Java Development

Selecting onshore Java developers eliminates the geographical divide altogether. With team members residing domestically in the same cities or states, alignment reaches new heights.

However, this approach sacrifices offshore and nearshore cost advantages. Other factors also come into play that technology leaders should carefully weigh.

### Unparalleled alignment

When Java developers work alongside internal stakeholders every day, seamless collaboration becomes possible:

- Effortless in-person meetings
- Rich personal connections

- Deep shared context

With onshore teams, miscommunications get cleared up instantly. And close partnerships between clients and developers fuel productive outcomes.

Despite best efforts, this level of seamless agility remains elusive in offshore and nearshore arrangements. The geographical divide—no matter how small—reduces critical high-bandwidth touchpoints.

Enhanced focus on innovation

According to a survey by PwC, over 90% of executives say innovation fuels revenue growth and competitive advantage. But real innovation relies on:

- Cross-functional collaboration
- Visibility into customer needs
- Understanding the prevailing business landscape

Onshore developers absorb this context intrinsically from their local environment. This allows them to zero in on creative solutions that move the needle rather than tactical outputs.

## Tighter Data Security

For industries like finance and healthcare, data protection represents an uncompromisable priority. In these cases, onshore teams may provide the only prudent option.

If developers work in the same legal jurisdiction as the rest of the organization, compliance controls remain straightforward. Local laws also



provide recourse in a worst-case incident.

Offshore regions with differing regulatory environments introduce harder-to-manage risks even with rigorous protocols.

## Access to Niche Capabilities

Organizations seeking leading-edge skills in emerging technologies often struggle to find these capabilities offshore—especially at scale.

Onshore developers generally have access to:

- Cutting-edge training/education options
- Visibility into the latest tools and trends
- Local professional networks to stay on the pulse

This exposure allows domestic teams to build expertise in bleeding-edge frameworks faster.

## Cost Factors

Of course, onshore Java teams deliver all these unmatched benefits at a price—literally.

Onshore developer salaries exceed offshore and nearshore rates.

However, take care not to underestimate the total cost of offshore arrangements as we discussed earlier.

Here's a comparison of estimated hourly rates across locations:

- India offshore developers: \$25–45 per hour
- Mexico nearshore developers: \$40–55 per hour
- United States onshore developers: \$65–126+ per hour

Rather than fixate solely on rate discrepancies, weigh the long-term value delivered by each model. Higher onshore rates could still drive better overall outcomes and ROI depending on the situation.

## Key Takeaways: Choosing the Right Java Team Model

So, across key factors like cost, communication, talent, data security and innovation, which option looks most appealing: offshore, nearshore or onshore?

The answer depends entirely on the organization and its specific goals and constraints. By carefully analyzing needs through the lens provided in this piece, technology leaders can determine the right strategic fit.

That said, a few best practices apply universally:

Adopt a context-driven mindset - Don't presume a single outsourcing arrangement (whether offshore, nearshore or onshore) can serve every project need across the organization. Different initiatives may call for different resourcing models based on variables like budget, risk tolerance, capabilities required and more.

Plan for hybrid teams - Blending external developers with internal resources can yield an optimal balance of cost efficiency and strategic alignment. Maintain internal oversight and knowledge sharing while leveraging vendor partnerships to scale faster.

Invest in collaboration infrastructure - With the right workflows, tools and protocols in place, distributed teams can overcome geographic divides. Prioritize solutions that bridge communication gaps regardless of developer location.

The worldwide talent pool offers limitless options for building quality Java teams. By focusing on sustainable value over pure cost savings, technology leaders can harness global resourcing models for maximum impact.

