How Do Stack Overflow Developers Live and Evolve in 2024?

Introduction

Every year, the Stack Overflow survey brings together voices from people who build software around the world. In this article we focus on three things everyone cares about: pay, education, and work experience. Our goal is to share clear, practical takeaways that help professionals, companies, and educators make better decisions—no code, no complicated terms, just plain language.

How We Keep The Study Clean

We focused on answers that were complete for the key questions. Many people skipped the pay and experience questions, so we only used responses that actually filled them in. This helps the results reflect reality and keeps the story honest and easy to trust.

Key Questions

Pay: How do incomes differ across people and places?

Education: How does school background relate to income?

Experience: How many years have people been building software?

Key Findings

1. Annual Compensation Distribution

The “middle” pay sits around **$110,000 USD**.

About **one in four** earns below roughly **$60,000 USD**, and about **one in four** earns above **$250,000 USD**.
What this means: The pay range is wide. Senior roles and big, expensive cities tend to pay the most.

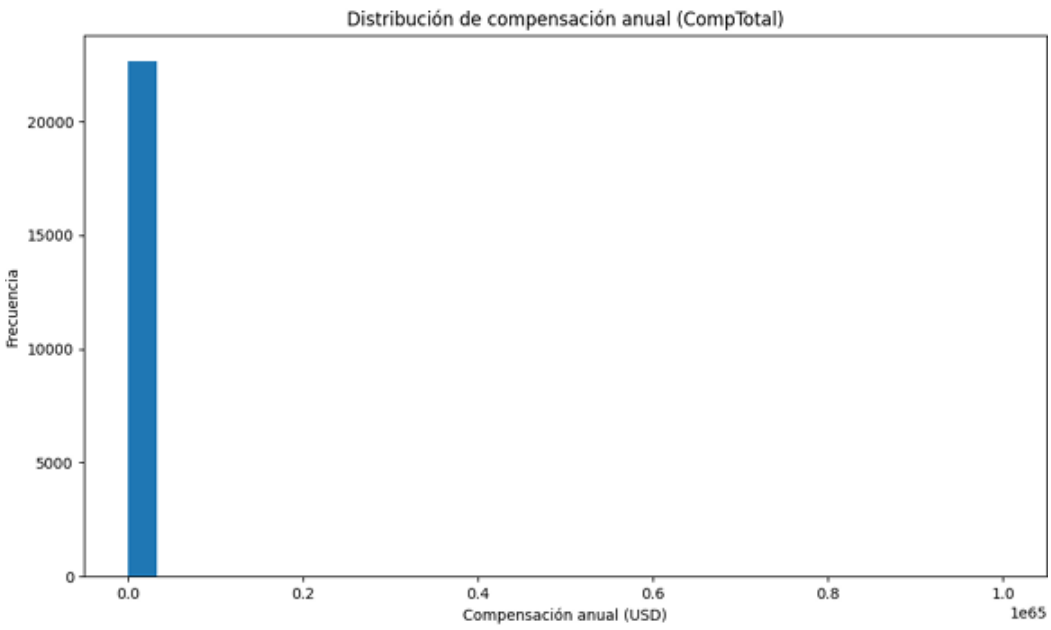

Figure 1. The pay chart spreads out a lot, showing a very mixed market.

2. Compensation by Education Level

Bachelor's degree: around $100,000 USD (middle).

Professional degrees (like PhD/MD/JD): around $90,000 USD.

Master's degree: around $80,000 USD.
What this means: Hands-on practice and in-demand skills often matter more for pay than a longer list of diplomas.

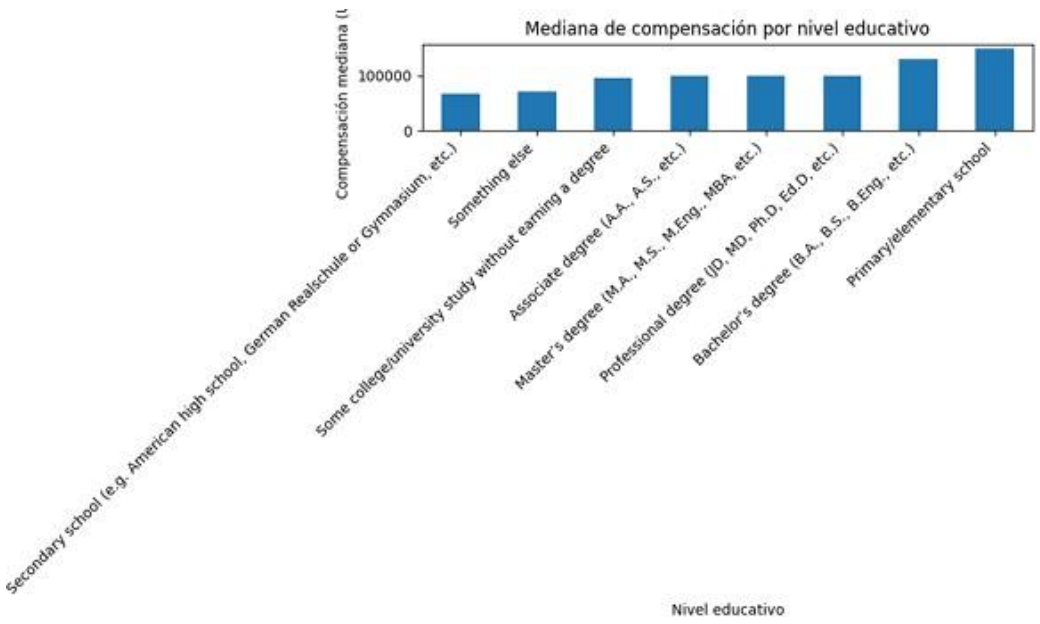

Figure 2. Side-by-side bars show how different education paths line up on pay.

3. Professional Trajectory (Years of Experience)

Many people report around **10 years** of experience.

A big group has **well over 10 years**, which helps teams with mentoring and stability.
What this means: There's a strong base of seasoned people who can guide others and keep projects on track.

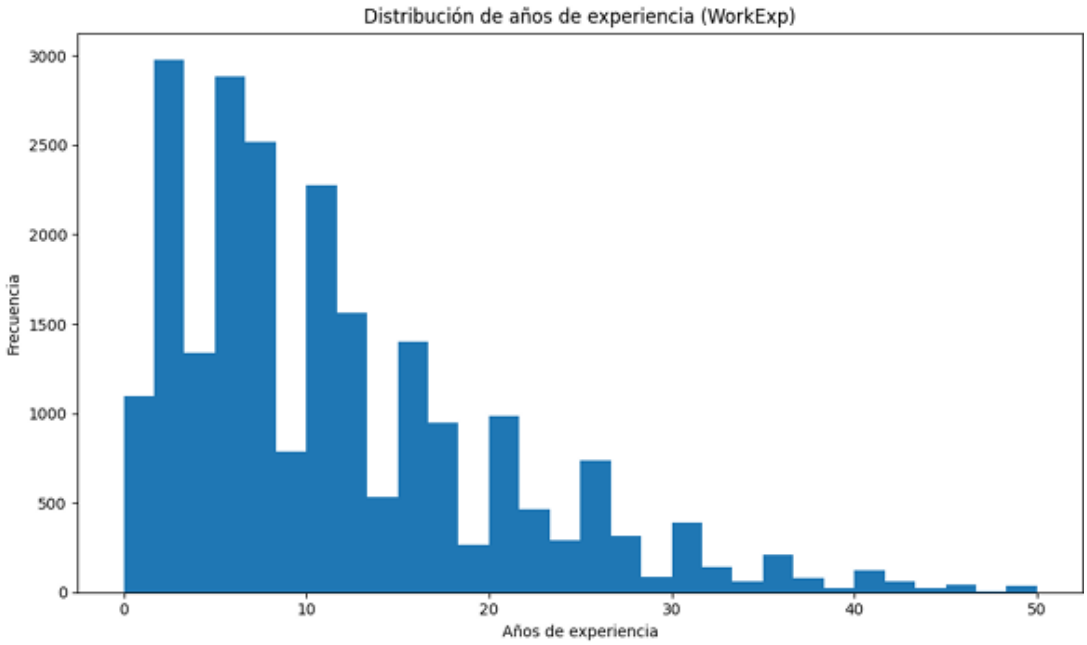

Figure 3. The experience chart shows lots of people with solid, long-term careers.

Modularized Code in src/utils.py

To ensure maintainability and follow DRY principles, we grouped functions into a documented module:

```
import pandas as pd

def cargar_datos(ruta_csv: str) -> pd.DataFrame:
    """
    Carga el CSV de Stack Overflow y limpia duplicados.
    - Elimina duplicados.
    - Filtra nulos en CompTotal, EdLevel y WorkExp.
    :param ruta_csv: Ruta al archivo CSV original.
    :return: DataFrame limpio listo para análisis.
    """
    df = pd.read_csv(ruta_csv, low_memory=False)
    df = df.drop_duplicates()
    df = df.dropna(subset=['CompTotal','EdLevel','WorkExp'])
    return df


def resumen_estadistico(df: pd.DataFrame, col: str) -> pd.Series:
    """
    Obtiene cuartiles y mediana de una columna numérica.
    :param df: DataFrame de análisis.
    :param col: Nombre de la columna.
    :return: Serie con Q1, mediana y Q3.
    """
```

```
    q1 = df[col].quantile(0.25)
    med = df[col].median()
    q3 = df[col].quantile(0.75)
    return pd.Series({'Q1': q1, 'Mediana': med, 'Q3': q3})
```

Conclusions and Recommendations

- **For Professionals**
- Build things you can **show**: small apps, features, or helpful tools. Results speak louder than buzzwords.
- Keep a short list of **skills employers look for** and grow them step by step.
- **For Companies**
- Offer clear **growth paths** and **mentoring** so people can advance with confidence.
- Support ongoing learning so teams stay sharp and motivated.
- **For Educators**
- Keep classes close to the **real world**: projects, teamwork, and practical problem-solving.
- Bring in voices from industry so students see how their work applies on the job.

Best Practices

- Use **simple charts** and short labels people can read at a glance.
- When you compare groups, include **straightforward numbers** that tell the story fast.
- Keep the text **short and friendly**; save the "how it was done" for a separate notebook.

Further Reading

- Articles on **writing for general audiences**.
- Guides to **telling stories with numbers**.
- Advice on **career growth** in tech.

Useful Links

- GitHub Repository
- Tools: pandas, matplotlib, scikit-learn
- Udacity Data Scientist Course

https://github.com/JuanPabloAR-ai/analisis-encuestas-stack-overflow