

NATIONAL UNIVERSITY OF THE ALTIPLANO PUNO

SCHOOL OF STATISTICS AND INFORMATICS



Course:

"SOFTWARE ENGINEERING I"

Teacher:

Torres-Cruz Fred

Students:

Mamani Mamani Ilma Magda

Puno - Perú

2024

Tasa de Actualización

Definición

La Tasa de Actualización es una métrica que mide la frecuencia con la que se actualiza el software en un período de tiempo determinado. Esta métrica es crucial para evaluar la agilidad del proceso de desarrollo y la capacidad del equipo para implementar mejoras, corregir errores y responder a las necesidades del usuario.

Importancia

- **Evaluación de la agilidad:** Permite medir la capacidad del equipo de desarrollo para entregar actualizaciones frecuentes.
- **Mejora continua:** Ayuda a identificar la rapidez con la que se pueden implementar mejoras y correcciones.
- **Satisfacción del usuario:** Una alta tasa de actualización puede indicar un equipo que responde rápidamente a las necesidades del usuario, mejorando la satisfacción.

Fórmula

$$\text{Tasa de Actualización} = \frac{\text{Número de actualizaciones en un período}}{\text{Período de tiempo}}$$

Ejemplo

Supongamos que en un proyecto de software se realizaron las siguientes actualizaciones en un mes:

- Actualización 1: 5 de junio
- Actualización 2: 12 de junio
- Actualización 3: 19 de junio
- Actualización 4: 26 de junio

Calculamos la Tasa de Actualización para el mes de junio (30 días) de la siguiente manera:

$$\text{Tasa de Actualización} = \frac{4}{30} = 0,133 \text{ actualizaciones por día}$$

Esto significa que, en promedio, se realiza una actualización aproximadamente cada 7.5 días.

Código en Python

A continuación, se presenta un código en Python que calcula la Tasa de Actualización basada en una lista de fechas de actualizaciones:

```
from datetime import datetime

def calcular_tasa_actualizacion(fechas_actualizaciones, periodo_dias):
    if not fechas_actualizaciones:
        return 0
    num_actualizaciones = len(fechas_actualizaciones)
    tasa_actualizacion = num_actualizaciones / periodo_dias
    return tasa_actualizacion

# Ejemplo de fechas de actualizaciones (formato: 'YYYY-MM-DD')
fechas_actualizaciones = [
    "2023-06-05",
    "2023-06-12",
    "2023-06-19",
    "2023-06-26"
]

# Convertir las fechas a objetos datetime
fechas_actualizaciones = [datetime.strptime(fecha, '%Y-%m-%d') for fecha in
fechas_actualizaciones]

# Período en días (para este ejemplo, usamos un mes de 30 días)
periodo_dias = 30

tasa_actualizacion = calcular_tasa_actualizacion(fechas_actualizaciones, periodo_dias)
print(f"La Tasa de Actualización es: {tasa_actualizacion:.3f} actualizaciones por día")
```

Código para Calcular la Tasa de Actualización

```
import csv
from datetime import datetime

def calcular_tasa_actualizacion(archivo_csv, periodo_dias):
    with open(archivo_csv, mode='r') as file:
        reader = csv.DictReader(file)
        fechas_actualizaciones = [datetime.strptime(row['fecha_actualizacion'], '%Y-%m-%d') for row
        in reader]

    if not fechas_actualizaciones:
        return 0
```

```
    num_actualizaciones = len(fechas_actualizaciones)
    tasa_actualizacion = num_actualizaciones / periodo_dias
    return tasa_actualizacion

# Nombre del archivo CSV
archivo_csv = 'actualizaciones.csv'

# Período en días (para este ejemplo, usamos un mes de 30 días)
periodo_dias = 30

tasa_actualizacion = calcular_tasa_actualizacion(archivo_csv, periodo_dias)
print(f"La Tasa de Actualización es: {tasa_actualizacion:.3f} actualizaciones por día")
```

Referencias Enlazadas

Aquí hay algunos enlaces a documentos importantes:

- [Software Updates as a Security Metric](#)
- [Software Metrics](#)