UNIVERSIDAD POLITÉCNICA DE TAPACHULA

DATOS GENERALES DEL PROCESO DE EVALUACIÓN

Nombre del alumno: Erick Colin Cruz Romero	Firma del alumno:
Matrícula: 213,34	Programa académico: IM
Título de dialogo: El Programación	Corte: Primer
Tipo de evidencia: Ordinaria	Periodo cuatrimestral: septiembre - Diciembre 23
Nombre del docente: LEI Wendy Karina López Lara	Firma docente:

Fecha: 26/09/23

INSTRUCCIONES

Revisar los documentos o actividades que se solicitan y marque en los apartados "SI" cuando la evidencia a evaluar se cumple; en caso contrario marque "NO". En la columna "OBSERVACIONES" ocúpela cuando tenga que hacer comentarios referentes a lo observado.

Valor del reactivo	Característica a cumplir (Reactivo)	Cumple SI	NO	Observaciones
5%	Puntualidad en la entrega de la presentación	✓		
5%	Tiempo de participación (3 min mínimo por alumno)	✓		
5%	Portada/hoja de presentación	✓		
50%	Desarrollo del diálogo a. Utiliza adecuadamente estructuras/elementos gramaticales. b. Se expresa con fluidez y pronuncia las palabras de manera correcta c. Interviene en la conversación de manera respetuosa empleando estrategias como "turn-taking" d. La conversación es coherente y guarda relación al tema tratado. e. Tiene dicción y utiliza un volumen de voz adecuado.	✓		
20%	Entrega de trabajo escrito físico incluye: 1. Introducción 2. Contenido 3. Conclusión	✓		
10%	Emplea vocabulario relacionado al tema de investigación tratado.	✓		
5%	Lenguaje corporal	✓		
	CALIFICACIÓN			

SEP

Documento controlado por medios electrónicos.
Para uso exclusivo de la Universidad Politécnica de Tapachula
www.uptapachula.edu.mx

SAC-RG.23
12 Jun 2015
Rev.01
Pág 1 de 5

UTyP

Ingeniería de Software

ED1. Realiza una conversación oral en la que se aborde el tema de literatura en el ámbito del estudio y en el tiempo libre. (parejas 5 min individual, trabajo escrito no script)

Nombre del alumno - Matrícula

Jorge Tronosco Vázquez Arendaño — 213317

Frick Colin Cruz Romero — 215184

Materia.

Inglés-7

Corte - tipo de evidencia

Primero – ordinaria

Cuatrimestre / periodo escolar

7° (septiembre. diciembre 2023)

Catedrático.

Dr. López Lara Wendy Karina

Tapachula, Chiapas; A 14 de septiembre de 2023

The Pragmatic Programmer: Your Code Should Tell You What It Does" is a classic book written by Dave Thomas and Andrew Hunt, originally published in 1999. This book has become an essential reference for programmers worldwide, offering timeless advice on how to write high-quality code, be a valuable team member, and stay current in an ever-evolving field.

Jorge: Hello Erick! How was your Software Engineering class today?

Erick: Hi Jorge! Honestly, it was quite interesting. Have you read any books on software or programming?

Jorge: Hi! I've been reading "The Pragmatic Programmer" by Dave Thomas, and I love it! It has a lot of valuable advice for improving my programming skills.

Erick: Hello! I'm glad to hear that you're enjoying the book. "The Pragmatic Programmer" is a classic in the programming world. Which pieces of advice or ideas have you found most interesting so far?

Jorge: Definitely, the idea of "Don't Repeat Yourself" (DRY) has caught my attention. I've always had a tendency to repeat code, and now I understand how important it is to avoid it.

Erick: That's an excellent lesson! Avoiding code repetition is crucial for maintaining clean and maintainable code. Have you been applying the DRY principle in your work or personal projects since you started reading the book?

Jorge: Yes, I've started refactoring some of my projects to eliminate code duplication. I've also been paying more attention to automating repetitive tasks, such as unit testing and project builds.

Erick: That sounds like great progress. Automation can save you a lot of time and effort in the long run. Are there any other concepts or chapters from the book that have interested you?

Jorge: I really liked the chapter on error handling. I learned that it's not always about avoiding errors but handling them properly and learning from them.

Erick: Exactly, learning to effectively handle errors is essential for writing quality software. Dave Thomas provides valuable advice on that topic in the book. Is there anything else you'd like to discuss or share about "The Pragmatic Programmer"?

Jorge: For now, that's what has stood out to me the most. I still have a lot to read, but I'm definitely applying what I've learned in my daily work!

Erick: That's great! Keep applying those principles and advice as you progress through the book. You'll surely continue to improve as a programmer. If you have more questions or want to discuss any other related topics, feel free to ask.

Jorge: Thank you! I definitely will.

Conclusion

The Pragmatic Programmer" has left an indelible mark on the programming industry since its publication in 1999. Its focus on pragmatism and efficiency in software development remains relevant and valuable today. This book has not only influenced generations of programmers but has also contributed to shaping programming culture, emphasizing the importance of continuous improvement, accountability, and attention to detail in the art of coding. As programming challenges evolve, "The Pragmatic Programmer remains an essential resource for programmers of all ages and experience levels. With its enduring wisdom, it continues to inspire and guide those who seek to excel in the world of software development.