

Extraído de una publicación de:



Autor Startup Guide IONOS



Fuente Startup Guide IONOS



Link <https://www.ionos.es/startupguide/productividad/modelo-en-espiral/>

Lectura 8

Modelo en espiral: el modelo para la gestión de riesgos en el desarrollo de software



El desarrollo de un nuevo software presenta grandes retos para todas las partes implicadas. Cuanto más compleja sea la aplicación que se desea desarrollar, más difícil resultará organizar de forma clara el proceso de desarrollo y controlarlo dentro de su complejidad. Por esta razón, normalmente nos apoyamos en planes especialmente diseñados paso a paso, denominados también modelos de procedimiento. Estos dividen el proceso de desarrollo completo en varias fases abarcables determinadas por el tiempo y el contenido. Uno de los modelos más conocidos, especialmente dirigido a la reducción de riesgos, es el denominado modelo en espiral, del año 1986.



¿Qué es el modelo en espiral?

Barry W. Boehm presentó su enfoque para el desarrollo de aplicaciones complejas en 1986 y en 1988 el ingeniero de software americano publicó su modelo en la publicación *A Spiral Model of Software Development and Enhancement* también en un contexto más general. En esta publicación, describía el modelo en espiral como una posible alternativa al modelo establecido hasta entonces, el modelo en cascada, que al mismo tiempo servía como base empírica. A diferencia del modelo en cascada, en el modelo en espiral no se parte de la base de que las tareas del desarrollo del software se organizan de forma lineal, sino que se entienden como tareas iterativas. Las fases no se realizan de forma única paso a paso, sino varias veces en forma de espiral. Mediante esta repetición cíclica, el proyecto se va acercando al objetivo de forma relativamente lenta, pero se minimiza de forma decisiva el riesgo de fracaso del proceso de desarrollo gracias a los controles regulares.

Definición

El desarrollo en espiral es un modelo de procedimiento para el desarrollo de software elaborado por Barry W. Boehm en el año 1986. Parte de la base de que el desarrollo de aplicaciones se debe llevar a cabo en un ciclo iterativo que se debe repetir tantas veces como sea necesario hasta alcanzar el objetivo. Gracias a las valoraciones regulares de los riesgos y a los controles rutinarios del producto intermedio, el modelo en espiral minimiza considerablemente el riesgo de fracaso en los proyectos de software.

Explicación del modelo en espiral: ¿cómo funciona el modelo en espiral?

Los problemas en el proceso de desarrollo pueden tener efectos muy diversos sobre el proyecto general. En cualquier caso, se debe contar con aumentos de costes, gastos adicionales y retrasos en el lanzamiento; factores que, especialmente para empresas pequeñas, pueden suponer un problema

existencial. Con su enfoque incremental e iterativo, que también contempla el análisis de riesgos periódico mediante diseños de prototipo, análisis o simulaciones, el modelo en espiral evita estos escenarios o al menos suaviza sus consecuencias negativas. El proyecto de software transcurre de forma continua hasta finalizar el ciclo del modelo en espiral, que principalmente abarca los cuatro pasos que aparecen a continuación.

Fase 1: definición de objetivos y alternativas y descripción de las condiciones generales

Un ciclo típico del modelo espiral comienza con la valoración de qué objetivos deben vincularse a cada uno de los pasos del desarrollo de software. Se puede tratar, por ejemplo, de la mejora del rendimiento o de la ampliación de la funcionalidad. Al mismo tiempo, es el momento de definir las alternativas para la implementación (por ejemplo, diseño A vs. diseño B) y determinar las condiciones generales como los costes o la inversión de tiempo.

Fase 2: valoración de las alternativas

En el siguiente paso, es hora de evaluar las alternativas, momento en el que los objetivos y las condiciones generales serán valores de referencia decisivos. En esta fase del ciclo del desarrollo en espiral, deberán identificarse los ámbitos de inseguridad que presenten un riesgo significativo para el progreso del proyecto de software. Después debe seguir la elaboración de las estrategias que presenten menos riesgo y que sean más rentables, para lo cual se podrán utilizar métodos como el modelo de prototipos, simulaciones, estudios comparativos, modelos de análisis y encuestas a usuarios.



Fase 3: desarrollo y revisión del resultado intermedio

Al finalizar el análisis de riesgos, se prosigue con el desarrollo real del software, así que esta fase siempre está caracterizada por los riesgos relativos restantes. Si el proceso de desarrollo está dominado por riesgos de rendimiento o de interfaz de usuario, o riesgos del control interno de la interfaz, se ofrece primero una estrategia de desarrollo evolutiva, donde se especifica el proyecto con más precisión y se optimizan los prototipos. El código real se escribe y se prueba varias veces hasta alcanzar el resultado deseado, que puede servir entonces como base de bajo riesgo para otros pasos de desarrollo.

Fase 4: planificación del siguiente ciclo

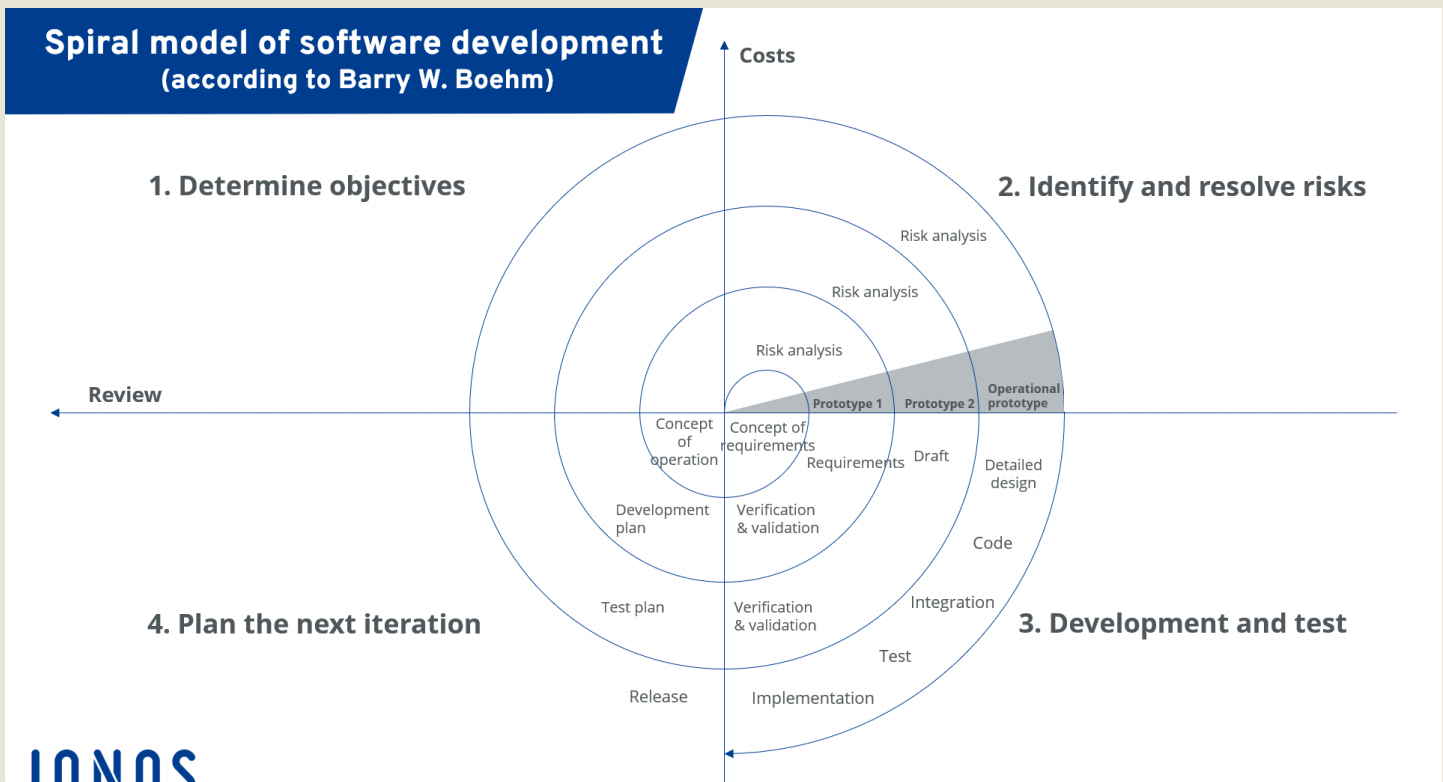
Una vez concluido un ciclo ya se empieza a planificar el siguiente ciclo. Por una parte, en forma de avance normal del proyecto, si los objetivos de un ciclo se han podido cumplir y se debe definir el siguiente objetivo. Por otra parte, también se puede tratar de encontrar soluciones, en caso de que la etapa de desarrollo anterior haya fracasado. En este caso, la estrategia seguida hasta entonces se puede sustituir, por ejemplo, con las alternativas definidas anteriormente o con una nueva alternativa. De esta forma, se puede intentar conseguir de nuevo el objetivo marcado.

Consejo

El modelo en espiral (desarrollo web) es un modelo de procedimiento genérico. Las cuatro fases solo presentan los objetivos fundamentales de un ciclo, pero no tienen que reflejarse en cada ciclo. Asimismo, en principio el modelo en espiral o desarrollo en espiral tampoco establece el orden de forma estricta. Por esta razón, el modelo se puede combinar en cualquier momento con otros métodos de procedimiento.

Representación gráfica del modelo espiral según Boehm

Una parte de la publicación del año 1988 es una representación gráfica del modelo en espiral que ejemplifica el aspecto del proceso de desarrollo web en forma espiral, basado en ciclos. Cada vuelta de la espiral representa en este esquema un ciclo completo, por lo que la hilera debe ajustarse siempre a cuatro cuadrantes distintos que, en este caso, se adaptan a las cuatro fases posibles del modelo. Cuanto mayor sea el tamaño de la espiral, mayor será el progreso obtenido, así como la aprobación de la revisión (eje X) y los costes de desarrollo (eje Y).



Ventajas y desventajas del modelo en espiral para desarrollo software

El desarrollo de software siguiendo el modelo en espiral es muy popular, sobre todo en proyectos grandes y complejos, en los que el control del presupuesto para promotores y empresas de desarrollo tiene especial importancia. En este caso, todas las partes implicadas se benefician del papel central del análisis de riesgo, que representa la mayor ventaja del desarrollo en espiral en comparación con otros modelos de procedimiento. La valoración periódica de los riesgos resulta especialmente ventajosa cuando se aplican entornos técnicos modernos que presentan, por norma general, un potencial de riesgo especial debido a la carencia de valores empíricos.

Asimismo, la estructura cíclica también es una de las ventajas del modelo; los conflictos entre diseño y exigencias técnicas que se yuxtaponen en el software se descartan casi por completo con los controles periódicos. Además, gracias al progreso en espiral, siempre es posible obtener y tener en cuenta el feedback. De esta forma, se pueden integrar a los promotores y a los usuarios en el proceso de desarrollo desde el principio. Sin embargo, un requisito imprescindible para poder disfrutar de estas ventajas es gestionar el proyecto activa y laboriosamente, controlando y documentando los ciclos individuales de forma continua y minuciosa.

Una prueba de que los numerosos pequeños pasos en los que se divide el desarrollo software según el modelo en espiral no siempre ofrecen ventajas es que, a pesar de los variados test, no es poco habitual que algunas partes inacabadas del programa se abran paso hasta el sistema productivo. En consecuencia, siempre existe el riesgo de que el producto final presente posibles errores o puntos débiles conceptuales. Además, en ocasiones se pueden dar retrasos en el desarrollo si en el transcurso de un ciclo o en la planificación del siguiente se deben tomar decisiones importantes que afecten al procedimiento posterior.

A continuación, se presentan las ventajas e inconvenientes del modelo espiral en forma de tabla:

Ventajas	Inconvenientes
Modelo flexible y genérico	Gran esfuerzo de gestión
Posible integración temprana de promotores y usuarios	Las decisiones periódicas pueden dilatar el proceso de desarrollo
Comprobaciones periódicas y enfocadas al riesgo	Hay errores e incongruencias conceptuales que se abren paso fácilmente al producto final a través del proceso de desarrollo desglosado
Conciliación perfecta entre exigencias técnicas y diseño	Know-how en análisis y gestión de riesgo esencial, pero no siempre disponible
Máximo control sobre los costes, recursos y la calidad del proyecto de software	No es apropiado para pequeños proyectos con un riesgo manejable
Apropiado para entornos técnicos novedosos	