3. CAPITULO III: RELACIÓN DE LAS TÉCNICAS PARA LA ELICITACIÓN DE REQUERIMIENTOS.

3.1. TÉCNICAS DE CAPTURA DE REQUERIMIENTOS

Los requerimientos tienen muchas fuentes en software típico, y es esencial que todas las fuentes potenciales estén identificadas y evaluadas para su impacto en él. Este asunto se diseña para promover el conocimiento de las varias fuentes de los requerimientos del software y de los armazones para manejarlos. Los puntos principales cubiertos son:

Metas: Son la motivación para el software, se debería prestar particular atención por parte de los ingenieros de software. Un estudio de viabilidad es una manera relativamente baja en costo de hacer esto.

Conocimiento del dominio: Permite deducir el conocimiento que los interesados no articulan y determinar las compensaciones que serán necesarias para requerimientos que están en conflicto.

Interesados o agentes de procesos: Los ingenieros que elaboran el software requieren identificar, manejar y representar puntos de vista de los diferentes tipos de clientes.

El entorno operacional: La toma de requerimientos serán derivados del ambiente donde el software será ejecutado.

El entorno de la organización: ingenieros desarrolladores deben ser sensibles al entorno organizacional su política y estructura, porque allí es donde servirá el software servirá de apoyo.

En realidad es un área muy difícil para los ingenieros de software, estos deben sensibilizarse al hecho de que por ejemplo los usuarios pueden tener dificultad para describir y anunciar sus tareas, pueden dejar la información importante sin especificar o con poca claridad, y en algunos casos poco dispuestos a colaborar. Es particularmente importante entender que la elicitación o captura de requerimientos no es una actividad pasiva y existen diferentes técnicas para realizarla, entre las principales están:

3.1.1. Entrevista

Es uno de los medios tradicionales utilizada para la elicitación de requerimientos y/o requerimientos. Debemos entender las ventajas y limitaciones de la entrevista y como deben ser conducidas. Las entrevistas son reuniones normalmente de dos personas, en las que se plantean una serie de preguntas para obtener las correspondientes respuestas en el contexto de un determinado **dominio** de problemas. En el ámbito de la ingeniería de requisitos, las entrevistas suelen realizarlas los ingenieros de requisitos al personal de la organización del cliente, con el objeto de abordar asuntos relacionados con los procesos de negocio o con características del software a desarrollar.

Las entrevistas deben **planificarse**, determinando su duración, los temas a tratar y seleccionando el número mínimo de personas a entrevistar dentro de la organización en función de su perfil profesional. En las primeras entrevistas los participantes suelen ser directivos que poseen una visión global de la organización y en ellas se abordan aspectos generales. Posteriormente, se suelen realizar entrevistas a los futuros usuarios para conocer las prácticas profesionales que llevan a cabo a diario y sus necesidades.

Es recomendable hacer llegar a los **participantes seleccionados**, con antelación a la entrevista, un resumen de los temas a tratar y los objetivos a alcanzar durante la misma. También puede ser útil enviar cuestionarios al objeto de situar al ingeniero de requisitos ante determinadas situaciones,

conocer la opinión de los usuarios sobre el sistema actual y sus expectativas ante el desarrollo del nuevo sistema.

3.1.2. Escenarios

El más común es el caso de uso, proporciona un marco para preguntas sobre tareas del usuario donde se realizan preguntas de “y si” y “como se hace esto”.

Componente	Descripción
Nombre	Identifica al escenario (en el caso de un sub-escenario, el título es el mismo que la sentencia episodio, sin las restricciones y/o excepciones.)
Objetivo	Establece la finalidad del escenario, a ser alcanzada en el contexto del problema. El escenario describe el logro del objetivo
Contexto	Describe las acciones previas necesarias para iniciar el escenario, las precondiciones, la ubicación física y temporal.
Recursos	Identifican los objetos pasivos con los cuales los actores trabajan.
Actores	Detalla las entidades que se involucran activamente en el escenario, es decir aquellas personas o estructuras organizacionales que tienen un papel en el escenario.
Set de episodios	Cada episodio representa una acción realizada por un actor, donde participan otros actores y se utilizan recursos. Los episodios se ejecutan secuencialmente. Un episodio también puede referenciar a un escenario. Su ocurrencia puede estar sujeta a condiciones. Se incluyen las *restricciones* del escenario o de cada episodio según corresponda.

Casos alternativos	Menciona los casos de excepción, que pueden corresponder a otros escenarios.

Tabla 2. Esquema de escenario

3.1.3. Prototipos (Prototipación)

Hay varias técnicas al respecto al prototipo, maqueta de versiones de pruebas beta de los diseños de pantalla del software, es una herramienta importante cuando se tienen que elicitar requerimientos o requerimientos confusos o complicados de interpretar. Un prototipo de requerimientos de software es una implementación parcial del sistema, se construye para ayudar a los desarrolladores, usuarios y clientes en la obtención de un mejor entendimiento del sistema, en especial de los requerimientos que están menos claros [Madigan].

En particular, uno puede sugerir que no se deberían construir prototipos a menos que realmente se piense que la construcción será rápida. Por otro lado, la prototipación debe realizarse solo cuando existe confianza mutua con el usuario [Davis: 2003]. La prototipación es una herramienta valiosa en la clarificación de requerimientos. Pueden actuar de manera similar en escenarios, al proveer a los usuarios un contexto en el cual se los puede entender mejor sobre la información que proveen. Existe una gran gama de técnicas de prototipado, desde borradores en papel de diseños de pantallas hasta versiones beta de los productos de software. Existe un fuerte solapamiento de su uso para la elicitación de requerimientos respecto a la validación de requerimientos [SWEBOK: 2004].

La prototipación es un medio común para la validación de la interpretación de los ingenieros de software de los requerimientos de software, así como también para la elicitación de los mismos. Dentro del proceso de elicitación de requerimientos, existe una amplia variedad de técnicas de prototipado. La ventaja de los prototipos es que hacen más fácil la interpretación de las

suposiciones de los ingenieros de software, y pueden poner en evidencia el porqué de equivocaciones. Por ejemplo, el comportamiento dinámico de una interfaz de usuario puede ser mejor entendida a través de un prototipo animado que a través de una descripción textual o modelos gráficos. También existen desventajas, dentro de las que se pueden nombrar: el peligro de que la atención del usuario se aparte de la funcionalidad principal por problemas cosméticos o problemas de calidad del prototipo. Una desventaja es que los prototipos pueden ser costosos de desarrollar. Sin embargo, si se evita el gasto incurrido al intentar satisfacer requisitos erróneos, su costo puede ser justificado muy fácilmente.

La prototipación ha sido muy útil para la elicitación de requerimientos, donde generalmente existe una gran cantidad de incertidumbre, o donde existe la necesidad de obtener una retroalimentación de los stakeholders. La prototipación usualmente se combina con otras técnicas, como por ejemplo al utilizar un prototipo para provocar una discusión sobre una técnica de elicitación en grupo, o como base para una encuesta.

3.1.4. Reuniones

Permite a través de grupos de gente interesados alcanzar un efecto aditivo y obtener más y mejor penetración en la elicitación de los requerimientos. Sin embargo, las reuniones necesitan ser correctamente dirigidas, y normalmente se requiere de un moderador o director. Favorecen la aparición de múltiples opiniones, creación, feedback y consenso colectivo.

3.1.5. Observación

Esta técnica es costosa por el tiempo utilizado para la captura de los requerimientos, los ingenieros de software aprenden sobre las tareas del usuario sumergiéndose en la observación de como un usuario obran o actúan recíprocamente con su software.