

## Trabajo de Fin de Grado

Grado en Ingeniería Informática

*Computación*

---

# **Diseño y desarrollo del módulo de incidencias multimedia de una plataforma BIM con realidad aumentada para la asistencia en la ejecución de operaciones de ingeniería y construcción**

---

*Jorge Salazar Beitia*

### **Dirección**

Borja Calvo Molinos  
Samara Ruiz Navarro

26 de junio de 2022



# Resumen

Este Trabajo Fin de Grado se enmarca en el desarrollo del proyecto *Tokii*, una plataforma que integra la metodología BIM (Building Information Modeling) con un Sistema de Información Geográfica y su visualización en Realidad Aumentada para la asistencia en la ejecución de operaciones de ingeniería y construcción.

La principal contribución de este trabajo al proyecto *Tokii* ha sido la incorporación de la creación y vinculación de contenido multimedia a las notas geoposicionadas, elementos que aportan información muy valiosa, directa e inmediata la hora de desarrollar un proyecto. Además, con el objetivo de mejorar la experiencia del usuario y mantener la concordancia de los elementos comunes del panel de control y de la aplicación se han realizado modificaciones gráficas en ambos.



# Índice de contenidos

Índice de contenidos	III
Índice de figuras	v
Índice de Diagramas	vii
<b>1 INTRODUCCIÓN</b>	<b>1</b>
1.1. Precedentes	2
1.2. Arquitectura digital	2
1.2.1. Maquetas Virtuales/BIM	3
1.2.2. AR+BIM	5
1.2.3. Geolocalización+AR+BIM	7
<b>2 EL PROYECTO TOKII</b>	<b>9</b>
2.1. ¿Qué es Tokii?	9
2.2. Panel de administración	11
2.2.1. Panel de control: Estadísticas de notas y geoposicionamiento de notas y modelos	11
2.2.2. Gestión de Proyectos	11
2.2.3. Subida y Posicionamiento de modelos	12
2.2.4. Gestión de Usuarios	13
2.3. Aplicación móvil	13
2.3.1. Menú principal	14
2.3.2. Menú de modelos 3D	15
2.3.3. El mapa	16
2.4. La API	17
<b>3 PLANIFICACIÓN</b>	<b>19</b>
3.1. Metodología de trabajo	19
3.1.1. Scrum	20
3.1.2. Documentación	21
3.1.3. Repositorios	22
3.2. Entorno de desarrollo	23
3.2.1. Unity	23
3.2.2. React	23
3.2.3. Material Design for Bootstrap (MDB)	24
3.3. Hitos del proyecto	24

<b>4</b>	<b>MEJORAS EN LA APLICACIÓN ANDROID</b>	<b>25</b>
4.1.	Hito 1: Creación de contenido multimedia enlazado a las notas . . . . .	25
4.1.1.	Toma de audio y toma vídeo . . . . .	25
4.1.2.	Creación de botones . . . . .	26
4.1.3.	Gestión y subida de archivos multimedia a la API . . . . .	28
4.1.4.	Nombrar archivos multimedia . . . . .	28
4.1.5.	Toma de audio, vídeo y foto 2.0 . . . . .	29
4.1.6.	<i>Path</i> y creación de contenido multimedia en la edición de notas . . . . .	30
4.2.	Hito 2: Descarga y muestreo del contenido multimedia . . . . .	31
4.2.1.	Pestaña de contenido multimedia . . . . .	31
4.2.2.	Botones de contenido multimedia y gestión de descargas . . . . .	31
4.2.3.	Muestra de contenido multimedia . . . . .	32
4.2.4.	Rotación de imágenes . . . . .	33
4.2.5.	Rotación de vídeos . . . . .	33
4.3.	Hito 3: Apariencia de las notas . . . . .	35
4.3.1.	Estado de la Nota . . . . .	35
4.3.2.	<i>Type</i> de la Nota . . . . .	35
<b>5</b>	<b>MEJORAS EN EL PANEL DE CONTROL</b>	<b>37</b>
5.1.	Hito 4: Modificación gráfica del Panel de Control . . . . .	37
5.1.1.	Modificación gráfica del login . . . . .	37
5.1.2.	Reestructuración del Panel de Control . . . . .	38
<b>6</b>	<b>CONCLUSIONES, TRABAJO FUTURO Y VALORACIÓN PERSONAL</b>	<b>45</b>
6.1.	Conclusiones . . . . .	45
6.2.	Trabajo futuro . . . . .	46
6.3.	Valoración personal . . . . .	46
	<b>Bibliografía</b>	<b>49</b>

# Índice de figuras

1.1.	Evolución de BIM desde 1974 hasta el siglo XXI. . . . .	4
1.2.	Dimensiones de BIM: funciones y software. . . . .	6
2.1.	Esquema de <i>Tokii</i> . . . . .	10
2.2.	Panel de Control. <i>Tokii</i> Manual de usuario . . . . .	11
2.3.	Detalle de un proyecto. <i>Tokii</i> Manual de usuario . . . . .	12
2.4.	Detalle de un Modelo. <i>Tokii</i> Manual de usuario . . . . .	12
2.5.	Detalle de un Usuario. <i>Tokii</i> Manual de usuario . . . . .	13
2.6.	Pasos de la Calibración. <i>Tokii</i> Manual de usuario . . . . .	14
2.7.	Añadir nota en dos pasos (posicionamiento y formulario). <i>Tokii</i> Manual de usuario . . . . .	15
2.8.	Sliders de transparencia y renderizado. <i>Tokii</i> Manual de usuario . . . . .	15
2.9.	Panel de edición del modelo. <i>Tokii</i> Manual de usuario . . . . .	16
2.10.	Visualización de notas geoposicionadas. <i>Tokii</i> Manual de usuario . . . . .	17
3.1.	Principios del Manifiesto Ágil . . . . .	19
3.2.	Elementos de Scrum . . . . .	21
3.3.	Captura de <i>Jira</i> . . . . .	22
3.4.	Ramas del <i>Bitbucket</i> utilizando <i>Git Flow</i> . . . . .	22
4.1.	Código base para convertir el audio en formato <i>Unity</i> a un formato de audio estándar . . . . .	26
4.2.	Comparación de botones . . . . .	27
4.3.	Inspector <i>Unity</i> . . . . .	27
4.4.	Sistema de archivos . . . . .	28
4.5.	Ventana de nombre . . . . .	29
4.6.	Sustitución <i>Regex.Replace</i> . . . . .	29
4.7.	Toma de multimedia . . . . .	30
4.8.	Pestaña de multimedia . . . . .	31
4.9.	Comparación de botones . . . . .	34
4.10.	Rotación vídeo . . . . .	34
4.11.	Cambio de estado . . . . .	35
4.12.	Cambio de tipo . . . . .	36
4.13.	<i>SpriteDataBase</i> . . . . .	36
5.1.	Cambio del Login . . . . .	37
5.2.	Cambio de cabeceras . . . . .	38
5.3.	Gráficas de incidencias . . . . .	39
5.4.	Incidencias geoposicionadas en el mapa . . . . .	40
5.5.	Ejemplo de <i>datatable</i> . . . . .	41

5.6. Tabla de proyectos y usuarios . . . . .	42
5.7. Algoritmo de ordenación de meses . . . . .	42
5.8. Comparativa entre el panel inicial y el final . . . . .	43



# Índice de diagramas

4.1. Diagrama de carga de contenido multimedia . . . . .	32
4.2. Diagrama de control de muestra de imagen y vídeo . . . . .	33



# INTRODUCCIÓN

La RAE define la arquitectura como «*Arte de proyectar, diseñar y construir edificios*». La plataforma *Tokii* es un proyecto en desarrollo de la empresa IMMERSIA Data Visualization que contribuye a esta labor en sus diferentes fases. *Tokii* reúne la metodología BIM (acrónimo de “Building Information Modeling”, en referencia al “Modelado de Información de la Construcción”) con un Sistema de Información Geográfica (en adelante GIS) y su visualización en Realidad Aumentada (en adelante AR) para la asistencia en la ejecución de operaciones de ingeniería y construcción.

Dentro del ciclo de vida del proyecto *Tokii* se han ido establecido una serie de hitos que marcan el progreso en su desarrollo. Cada hito tiene asignadas una o varias personas responsables de su ejecución y se vincula a diferentes tareas para superarlo.

Este Trabajo Fin de Grado se enmarca en el desarrollo del proyecto *Tokii* y en él se presentan los diferentes hitos en los que ha participado el autor del mismo como miembro del equipo de trabajo, realizando todas las tareas vinculadas a los mismos. El trabajo se ha llevado a cabo a lo largo de 4 meses en el marco de un Convenio para el Desarrollo de un Programa de Cooperación Educativa y de un Convenio de Colaboración para la Elaboración y Dirección de Trabajos Fin de Grado entre la Facultad de Informática de la UPV/EHU y la empresa IMMERSIA Data Visualization, de 300 horas de duración cada uno.

La realización de las tareas vinculadas a los diferentes hitos se ha llevado a cabo siguiendo metodologías ágiles, concretamente *Scrum*, y para gestionar *Scrum* se ha utilizado *Jira*, un software que está diseñado para planificar, supervisar y publicar software de calidad asegurada. *Jira* se enlaza con el espacio *Confluence* donde se aloja toda la documentación del proyecto. La gestión del código del proyecto *Tokii* se ha realizado utilizando *Bitbucket*.

En lo que respecta al entorno de desarrollo vinculado a los hitos en los que se ha trabajado, se ha utilizado *Unity* para lograr la integración de BIM con AR en la aplicación y *React* y *MDB* para el funcionamiento y el diseño del panel de control respectivamente.

A lo largo de este Trabajo Fin de Grado se expondrán la metodología de trabajo, el entorno de desarrollo vinculados a los diferentes hitos y tareas en las que se ha trabajado para el logro de estos hitos. Así, el trabajo consta de varios capítulos. En el presente capítulo se hace un breve repaso de la evolución de la arquitectura desde la llegada de los ordenadores hasta la actualidad y se presentan algunas de las innovaciones tecnológicas, que incorpora

## 1. INTRODUCCIÓN

---

la plataforma *Tokii*, que están haciendo posible la digitalización del sector de la construcción. Posteriormente, en el capítulo 2 se presenta el proyecto *Tokii* y las principales características y funcionalidades del mismo. En el capítulo 3 se presenta la metodología de trabajo y el entorno de desarrollo del proyecto. En los capítulos 4 y 5 se detalla cuáles son las tareas realizadas por el autor de este Trabajo Fin de Grado en los hitos del proyecto que se le han asignado como miembro del equipo de desarrollo. Para finalizar, en el capítulo 6 se exponen las principales conclusiones del trabajo, el trabajo futuro y una valoración personal de lo que ha supuesto para el autor del mismo, tanto desde un punto de vista personal y como profesional, el haber tenido la oportunidad de contribuir al desarrollo del proyecto.

### 1.1. Precedentes

Para comprender la contribución de la metodología y las herramientas en las que se basa la plataforma *Tokii* a la arquitectura actual, es de utilidad conocer cómo hemos llegado hasta aquí, cómo se ha pasado del lápiz y el papel a las tecnologías más innovadoras.

El hombre vivió durante mucho tiempo protegiéndose en refugios naturales. Es en el Neolítico, con la aparición del sedentarismo y las primeras herramientas cuando comienza a edificar sus propias construcciones para sus actividades rituales y domésticas. Es así como nace la arquitectura y son la cabaña (construcción civil) y el menhir (construcción religiosa) sus primeras manifestaciones.

A lo largo de la historia la evolución de la arquitectura ha estado ligada a la evolución social, cultural y tecnológica de la sociedad. Si bien los aspectos sociales y culturales definen, sobre todo, los diferentes estilos arquitectónicos y su evolución, han sido los cambios tecnológicos (nuevos procedimientos, nuevas herramientas y nuevos materiales) los que han determinado en gran medida las características estéticas y funcionales de las construcciones.

Hasta hace aproximadamente 50 años, los grandes avances tecnológicos en arquitectura se han basado en la aparición de nuevos materiales (vidrio, hierro, acero, hormigón armado, etc.) e innovaciones tecnológicas (la máquina de vapor, la electricidad, el mecanizado, etc.) vinculadas principalmente a la construcción. A finales de la década de 1970, con la llamada "Revolución Digital" se producen cambios en todos los ámbitos. En arquitectura, la incorporación de la informática en el proceso de diseño supone el comienzo de una nueva etapa, la arquitectura digital, y da comienzo a la transición de lo analógico a lo digital.

### 1.2. Arquitectura digital

Con la llegada de los ordenadores, la arquitectura comienza su proceso de digitalización. En la década de 1980 se ofrecen las primeras soluciones de software para diseño 2D asistido por ordenador, el CAD (siglas en inglés de Computer Aided Design). Se trata de la herramienta de construcción más básica de cualquier dibujo técnico y se utiliza, fundamentalmente, para la creación de planos 2D y dibujos lineales de una forma rápida y precisa. Sus versiones más avanzadas ofrecen también capacidades 3D y se siguen utilizando en la actualidad. La tecnología CAD es, sin embargo, una forma de diseño tradicional que reproduce la técnica del papel y el lápiz sustituyendo el papel por la pantalla de ordenador y el lápiz por el ratón.

Con el tiempo las herramientas 2D alcanzan las tres dimensiones con el empleo de

modelados 3D y renders que permiten dar vida a los proyectos antes de que sean reales. Gracias a la modelización 3D y la posterior renderización se pueden presentar infografías 3D con un gran impacto visual y realismo.

En la actualidad, con la ayuda de estas tecnologías, los procesos de representación gráfica están muy avanzados y existen infinidad de soluciones de software para el diseño 2D y el modelado y renderizado 3D. Sin embargo, el proceso de digitalización en la arquitectura va más allá de la creación de espacios y objetos. La digitalización, a través de la incorporación de las nuevas tecnologías, debe ser una oportunidad que las empresas de la construcción deben aprovechar para adaptarse al presente y evolucionar.

A continuación se presentan algunas de las innovaciones tecnológicas más importantes que están haciendo posible la digitalización del sector de la construcción y que incorpora el proyecto *Tokii*.

### 1.2.1. Maquetas Virtuales/BIM

La Asociación BuildingSMART Spain<sup>1</sup> define BIM como «una metodología de trabajo colaborativa para la creación y gestión de un proyecto de construcción. Su objetivo es centralizar toda la información del proyecto en un modelo de información digital creado por todos sus agentes»[1].

La metodología BIM tiene sus orígenes en el denominado “Sistema de Descripción de Edificaciones”( en inglés, Building Description System o BDS) introducido por Charles M. Eastman, la primera persona que realiza un modelo virtual inteligente de un edificio en 1974. Eastman et al. [2] enumeran las principales problemas de la arquitectura tradicional que pueden solucionarse con estos modelos:

- Información redundante, que debe replicarse.
- Representación de las construcciones (de tres dimensiones) en 2D, que hace que sea necesario plasmar al menos dos caras.
- Dificultades para mantener la información al día.
- Representaciones que no incorporan información numérica, que debe añadirse aparte.
- Planos que no evolucionan con el edificio, por lo que quedan obsoletos.

En el Abstract del artículo de 1974 puede leerse: «*Muchos de los costes de diseño, realización y funcionamiento de las construcciones derivan del hecho de la confianza los planos para describir toda la información de un edificio. Como alternativa, este documento propone un diseño de un sistema informático útil para almacenar y manipular la información de un proyecto con un detalle que permite el diseño, la construcción y el análisis operativo. Un edificio se considera como la composición espacial de un conjunto de elementos.*». Así proponen el sistema BDS que tiene las siguientes características:

---

<sup>1</sup>BuildingSMART Spanish es una asociación sin ánimo de lucro formada por todos los agentes del sector de la construcción cuyo principal objetivo es fomentar la eficacia en el sector de la construcción a través del uso de estándares abiertos de interoperabilidad sobre BIM para alcanzar nuevos niveles en reducción de costes y tiempos de ejecución y aumento de la calidad.

## 1. INTRODUCCIÓN

1. Facilita la inserción gráfica de formas y elementos complejos.
2. Tiene un lenguaje gráfico interactivo.
3. Permite producir perspectiva o dibujos ortográficos de alta calidad.
4. Permite clasificar las base de datos por atributos.

Tal y como se refleja en la Figura 1.1[3], a partir de esta investigación se abre un nuevo escenario para la industria de la construcción y el BDS evoluciona hasta convertirse en lo que actualmente denominamos BIM, gracias innovaciones tecnológicas que mejoran la precisión en el diseño e introducen un mayor número de funciones que permiten lograr un mayor grado de gestión.

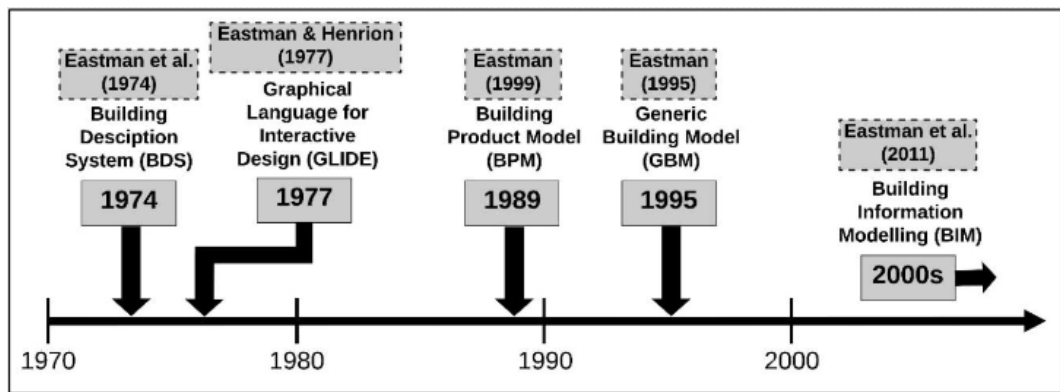


Figura 1.1: Evolución de BIM desde 1974 hasta el siglo XXI.

BIM nos permite obtener digitalmente un modelo virtual muy preciso de una construcción. Aunque para construir un modelo tridimensional es suficiente con tres dimensiones (3D), cuando hablamos de BIM nos referimos a una metodología 7D. Las 7 dimensiones que se distinguen en BIM son [4]:

- **1D-IDEA:** Fase previa y datos iniciales (ubicación, análisis de suelo, permisos, etc.).
- **2D-BOCETO:** Caracterización básica del proyecto y preparación del software para modelar.
- **3D-MODELO TRIDIMENSIONAL:** modelo en formato 3D basándose en la información y las características del proyecto definidas en las dimensiones anteriores mediante el uso de animaciones o renders.
- **4D-PLANIFICACIÓN:** definición de las fases del proyecto, planificación temporal que incluye simulaciones sobre diferentes parámetros que pueden afectar a la construcción.
- **5D-COSTES:** Estimación y análisis de costes y obtención de informes presupuestarios.
- **6D-ANÁLISIS DE SOSTENIBILIDAD:** simulación de las alternativas posibles y búsqueda de la alternativa óptima del proyecto en términos de sostenibilidad.

- **7D-GESTIÓN DEL CICLO DE VIDA:** organización de la información relevante del proyecto y de las recomendaciones y normas a seguir para su uso y mantenimiento durante la vida del proyecto.

Mediante BIM se crea, por lo tanto, un modelo 3D virtual del edificio que se va a construir. Todos los elementos que lo forman tienen las características físicas de sus componentes reales; es decir, son prototipos digitales de todos los elementos de los que está compuesto el edificio. Toda la información relevante sobre cada elemento está recogida en una única base de datos, no existen archivos duplicados, y todos los agentes que intervienen en el proceso de construcción pueden intercambiar información de manera coherente mejorando así la comunicación. A partir de este modelo 3D se obtiene una maqueta virtual del edificio que representa el proyecto de construcción completo, incluyendo la ubicación, orientación y las características de todos los elementos. Esto permite saber el comportamiento de que tendrá el edificio y sus componentes en diferentes situaciones sin necesidad de estar construido. Permite, además, realizar cálculos estructurales y de instalaciones, análisis energéticos, obtener planos y calcular los costes, lo que lleva a una gestión integral óptima del edificio.

Es importante señalar que BIM no es un software, pero, al tratarse de una metodología que permite gestionar y compartir datos y modelos, almacenar y estandarizar información, obtener un resultado óptimo en la representación del modelo 3D, etc., necesita software para poder desarrollarse. Estos softwares son conocidos como softwares BIM y soportan las principales dimensiones de BIM que se han señalado previamente. Existe una gran variedad de empresas de software que llevan tiempo desarrollándolos y comercializándolos[5]. En la Figura 1.2 se muestran algunos de los más utilizados en cada fase.

BIM es, por lo tanto, una herramienta que permite a todos los agentes implicados en el proyecto trabajar de una forma colaborativa para construir de una manera más eficiente y sostenible y reduciendo costes. El Informe del Observatorio Europeo de la Construcción de marzo de 2019 estima que el uso de la metodología BIM supone un ahorro de costes entre el 13 % y el 21 % en la fase de construcción y entre el 10 % y el 17 % en la fase de operaciones [6].

Sin embargo, a pesar de que desde las Administraciones Públicas se ha impulsado la implantación progresiva de BIM, el proceso está siendo lento. La Comisión BIM, impulsada por el Ministerio de Fomento, en el documento titulado “BIM en 8 puntos. Todo lo que necesitas conocer sobre BIM”[7] señala: «*El sector de la construcción ha llegado hasta el s. XXI sin haber integrado eficazmente los avances tecnológicos, de innovación y de gestión que le permitan tener similares niveles de productividad y competitividad que el resto de sectores industriales*». En este sentido, existen una serie de barreras que impiden el avance en su implantación [8] que se pueden clasificar en cinco grupos: tecnología, coste, administración, personal y factores legales. No obstante, teniendo en cuenta que BIM posibilita la integración de otras soluciones de tecnologías de la información y la comunicación que pueden aplicarse directamente a la construcción, como son la realidad virtual (en adelante VR) y la AR, los GIS, el uso de drones y de la robótica, etc. la incorporación de las mismas puede contribuir a la eliminación de las barreras y, como consecuencia, a la digitalización del sector.

### 1.2.2. AR+BIM

Carmigniani et al. [9] definen AR «*como una vista directa o indirecta en tiempo real de un entorno físico del mundo real que ha sido mejorado/aumentado al agregarle información*

## 1. INTRODUCCIÓN



**Figura 1.2:** Dimensiones de BIM: funciones y software.

*virtual generada por computadora*». AR es, por lo tanto una tecnología inmersiva, interactiva y registrada en 3D que combina objetos reales y virtuales. La AR genera un entorno en el que se superponen el mundo real y objetos virtuales y/o información, lo que da como resultado un mundo mixto. Normalmente, la AR utiliza la cámara de un dispositivo móvil o una gafas de material transparente y permite al usuario interactuar con el mundo real sin bloquear el mundo físico.

La posibilidad de añadir objetos virtuales e información al entorno a través de la AR, la hace interesante para aplicar en muchos campos, también en el sector de la construcción. Safikhani et al. [10] señalan que AR, junto con la realidad virtual (VR) y la realidad mixta (XR)<sup>2</sup>, puede considerarse una de las tecnologías prometedoras que pueden mejorar el proceso de diseño, construcción, aplicación y mantenimiento en muchos proyectos. En este sentido, Saar et al. [3] señalan que la integración de BIM con AR puede mejorar su uso y su aplicabilidad, principalmente, en el trabajo de campo, debido al limitado nivel de interacción entre el mundo virtual (el modelo) y el real (la situación física) que tiene la metodología BIM por sí sola. En este sentido, el uso de la AR en BIM aporta una serie de ventajas:

- Permite visualizar e interactuar con el modelo BIM in situ, en el lugar de construcción,

<sup>2</sup>En contraste con AR, la VR es una experiencia de inmersión completa en un entorno artificial simulado y la RX combina el mundo real con un mundo tridimensional artificial



de una forma natural y similar a como se haría con un edificio o con una maqueta en el mundo real.

- Permite incorporar y compartir información cuando y donde se necesita.
- Mejora la visualización y comprensión de los datos.
- Mejora la detección de fallos en los elementos arquitectónicos o estructurales.
- Permite la simulación de procesos y la monitorización y la supervisión en tiempo real, lo que mejora la seguridad y la calidad en las diferentes fases del proyecto.
- Mejora la comunicación entre todos los agentes que interviene en el proyecto desde la fase de diseño hasta la de gestión de ciclo de vida del edificio una vez construido.
- Favorece la motivación y comprensión, lo que lleva a un aprendizaje más rápido y natural.

Todas estas ventajas ponen de manifiesto que la integración de BIM con AR puede contribuir a la implantación progresiva de esta metodología y, por lo tanto, a la digitalización y modernización del sector de la construcción.

### 1.2.3. Geolocalización+AR+BIM

A la AR se puede añadir el uso de la geolocalización, lo que permitirá obtener la ubicación geográfica real de un objeto dentro de la imagen. Esto se lleva a cabo gracias a la combinación de AR y los GIS.

Los GIS hacen referencia a un conjunto de herramientas para el procesamiento de datos geográficos. Son responsables de integrar varios aspectos de los datos geográficos del mundo real y recopilar, operar, analizar y visualizar estos datos. Los GIS tiene múltiples aplicaciones porque permiten trabajar con todo tipo de datos geográficos del mundo real: datos directamente relacionados con la tierra, como datos topográficos y geológicos, datos estadísticos y epidemiológicos relacionados con información demográfica, la actividad humana y la actividad económica, etc. Los GIS utilizan la información con referencias espaciales de muchas fuentes, incluyendo fotos aéreas, imágenes de satélite y GPS. También se pueden usar de manera efectiva para producir mapas de la tierra que combinan varias capas de datos geográficos; el ejemplo más conocido de tal GIS es Google Earth [11].

La integración de BIM con AR y GIS añade, a las ventajas que se señalaban en el apartado anterior, la posibilidad de determinar la posición de un objeto superpuesta sobre sus coordenadas geográficas en la pantalla de los equipos de AR y combinar la información geométrica y semántica, propia del BIM, con información geográfica [12].

Sin embargo, el geoposicionamiento en AR lleva consigo una serie de desafíos a nivel técnico en relación al modelo BIM que hay que afrontar:

- Precisión: la posición de un punto del modelo en AR debe coincidir con la posición establecida teóricamente en el mismo.
- Estabilidad: Al mover el dispositivo, el modelo geoposicionado debe permanecer en su lugar geográfico.

## 1. INTRODUCCIÓN

---

- Orientación: el modelo debe girar de acuerdo al giro del dispositivo.

En los últimos años, se han creado aplicaciones que aúnan la metodología BIM con la AR y el geoposicionamiento, como es el caso del proyecto *Tokii* que se presenta a continuación.

## EL PROYECTO TOKII

### 2.1. ¿Qué es Tokii?

Las plataformas de BIM líderes en el mercado, como Synchro y Autodesk Naviswork, permiten planificar un proyecto y asociarlo a un modelo virtual en Autodesk Revit o en ArchiCAD (Dimensiones 1 a 3), realizar un seguimiento temporal de la obra en tiempo real y comparar lo desarrollado con lo planificado (Dimensión 4) y llevar a cabo una estimación y análisis de costes, obteniendo los informes presupuestarios (Dimensión 5). Además, existen diversos *plugins* para visualizar el modelo y los datos en VR, AR o mixta en estos programas (Enscape, Augment, Synchro XR, etc.). Sin embargo, estas herramientas requieren un tiempo muy elevado para desarrollar y renderizar el modelo BIM y tienen baja calidad en los gráficos y retraso en los fotogramas. Todo esto se traduce en una falta de realismo en el entorno de AR. Además, ninguno de ellos permite obtener información geoposicionada.

En los últimos años se han desarrollado diversas aplicaciones, entre ellas la plataforma *Tokii*, que integran BIN con AR y GIS y que permiten desarrollar de forma óptima las 7 dimensiones de BIM.

La plataforma *Tokii* es una propuesta de la empresa IMMERSIA que «reúne la metodología BIM con la información GIS y su visualización en AR, mediante herramientas avanzadas de visualización 3D y técnicas de inteligencia artificial, para integrar y analizar toda la información generada a lo largo de su ciclo de vida (desde la planificación hasta el derrumbe). En este sentido, se abre la puerta a la dimensión 7D dentro de la metodología BIM en un entorno inmersivo gracias a la aplicación de los gemelos digitales[13].»

El proyecto *Tokii* consta principalmente de tres elementos: un panel de administración, una API y una aplicación multidispositivo. Estos tres elementos se conectan entre sí para poder lograr la experiencia en AR de la vista de modelos y incidencias o notas geoposicionadas de manera exacta (Figura 2.1). Todo esto se logra a través de la aplicación con el uso de la cámara y la posición dada por una antena GNSS <sup>1</sup> que está conectada al dispositivo y que

---

<sup>1</sup>GNSS, por sus siglas en inglés Global Navigation Satellite System, hace referencia a todos los sistemas de navegación basados en satélites (GPS, GLONASS, Galileo, Compass, etc.) que proporcionan señales desde el espacio. Estos sistemas transmiten datos de posicionamiento a los receptores a través de una antena. Posteriormente, los receptores usan estos datos para determinar la ubicación.[14]

## 2. EL PROYECTO TOKII

sirve de gran ayuda a los arquitectos en la proyección y la edificación de proyectos ya que permite obtener la vista de los mismos construida en AR. Sus principales funcionalidades son las siguientes:

- Catálogo de modelos 3D y visualización geoposicionada en AR.
- Anotaciones geoposicionadas en AR y mapa.
- Estado de construcción o avance de la obra en tiempo real.
- Creación de gemelo digital del edificio tras la finalización de la obra.

En la siguiente sección, se describe con más detalle los elementos principales del proyecto *Tokii* y las funcionalidades del mismo. Para esta descripción se ha utilizado el Manual de Usuario elaborado por IMMERSIA y el conocimiento personal del proyecto.

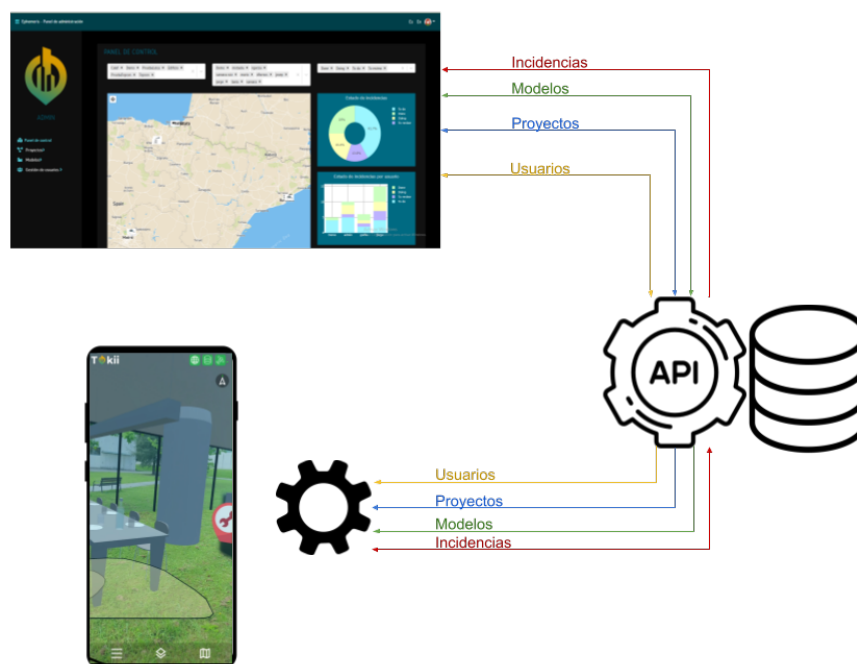


Figura 2.1: Esquema de *Tokii*

Antes de realizar esta descripción, es conveniente definir dos de los elementos principales de un proyecto desarrollado con *Tokii*:

- Un modelo 3D de una construcción es una representación tridimensional de la misma a través de un software que utiliza una serie de datos matemáticos y geométricos. Estos modelos son realizados por los arquitectos e integran toda la información que contienen los planos de una construcción (elementos arquitectónicos, instalaciones, etc.). Como se ha mencionado previamente, las plataformas de BIM permiten obtener este tipo de modelos.
- Una nota geoposicionada contiene información relevante de un punto geográfico. En el caso de la plataforma *Tokii*, las notas van asociadas a modelos y proyectos y contiene la información que se puede considerar útil para el desarrollo de los mismos.

Un proyecto desarrollado con *Tokii* se compondrá, por lo tanto, de un conjunto de modelos y notas, por ejemplo, el conjunto de modelos 3D de una obra a realizar y todas las notas geoposicionadas que nos dan información relevante para el proyecto.

## 2.2. Panel de administración

El panel de administración es una plataforma accesible vía navegador que permite la gestión de los diferentes contenidos y permisos que se visualizarán a través de la aplicación móvil. Así mismo, permite contemplar y gestionar todo lo relacionado con los proyectos creados.

Los contenidos del panel de administración son el panel de control, los proyectos, los modelos y la gestión de usuarios. A continuación se describe cada uno de ellos.

### 2.2.1. Panel de control: Estadísticas de notas y geoposicionamiento de notas y modelos

El panel de control es la pantalla inicial y punto de partida de la aplicación donde se presenta un mapa interactivo con los modelos y notas previamente geoposicionadas (Figura 2.2). Cada modelo o nota contiene una descripción de dicho objeto y las notas contienen también estadísticas sobre su estado, así como de los usuarios asignados a ellas.

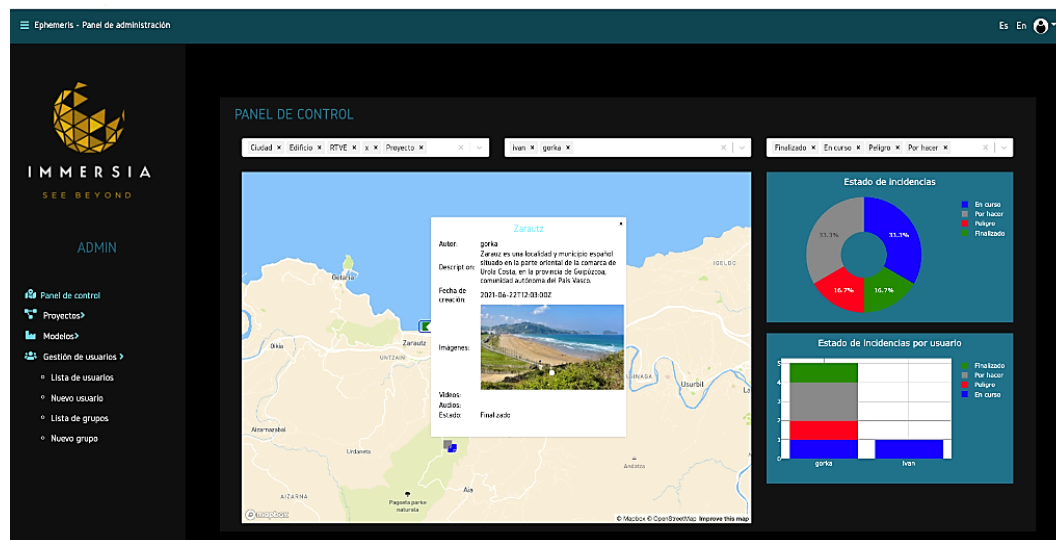


Figura 2.2: Panel de Control. Tokii Manual de usuario

### 2.2.2. Gestión de Proyectos

Como se ha mencionado anteriormente, un proyecto es un conjunto de modelos y notas, a los que, de acuerdo a los permisos de usuario correspondientes, un grupo de usuarios tiene acceso de manera ordenada. Cuando se crea un nuevo proyecto, se le asigna un nombre y una descripción (opcional) y se gestiona el acceso de usuarios y de grupos de usuarios a las notas y a los modelos que contiene. Cada proyecto contiene información sobre los grupos de usuarios que tienen acceso al mismo y los modelos que incluye (Figura 2.3).

## 2. EL PROYECTO TOKII

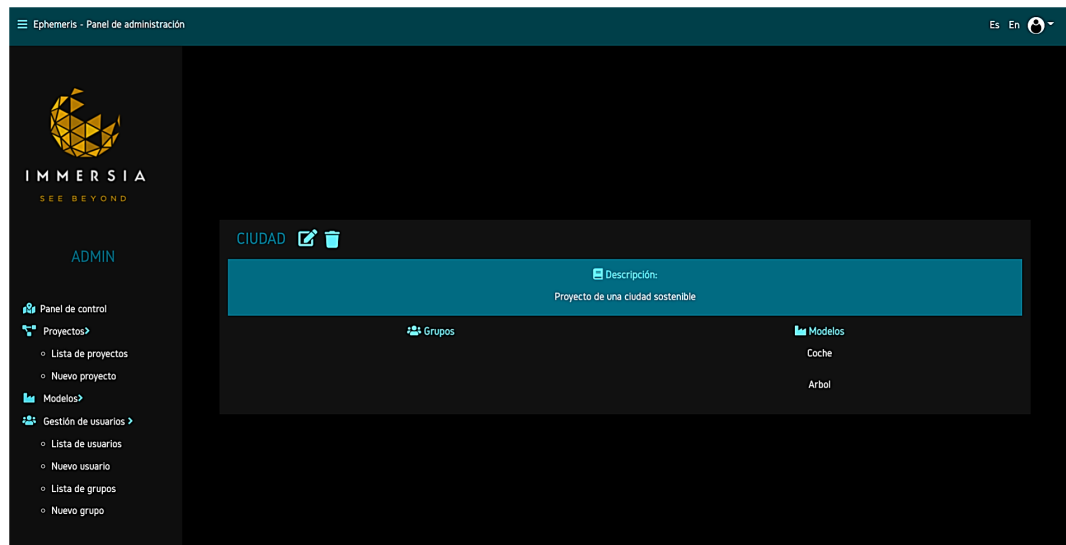


Figura 2.3: Detalle de un proyecto. Tokii Manual de usuario

### 2.2.3. Subida y Posicionamiento de modelos

El módulo de Modelos permite subir, editar y visualizar los distintos modelos subidos por los grupos a los que el usuario tenga acceso. Tenemos distintas variables a completar que representan las distintas opciones: establecer el modo de gestionar, asignar proyecto, subir las distintas capas del ifc (formato BIM) y geoposicionarlo y darle un nombre y una descripción (Figura 2.4).

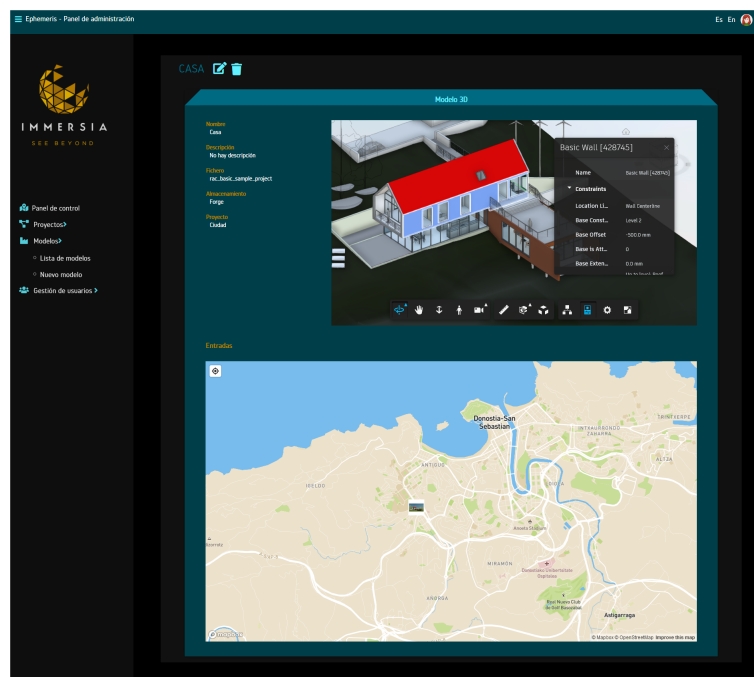


Figura 2.4: Detalle de un Modelo. Tokii Manual de usuario

### 2.2.4. Gestión de Usuarios

En el módulo Gestión de Usuarios el administrador podrá obtener información sobre todos los usuarios, el grupo al que pertenecen y los permisos que tienen (Figura 2.5). Así mismo, puede visualizar el listado de permisos que tiene un grupo determinado para el acceso a los distintos modelos, sus proyectos asociados y los usuarios pertenecientes al grupo. Además, puede crear nuevos usuarios o grupos, eliminarlos o modificar sus permisos. Si se borra un usuario o grupo, permanecerá desactivado en el sistema y perderá el acceso al mismo, pero su información quedará asociada a las notas de las cuales ha sido responsable.

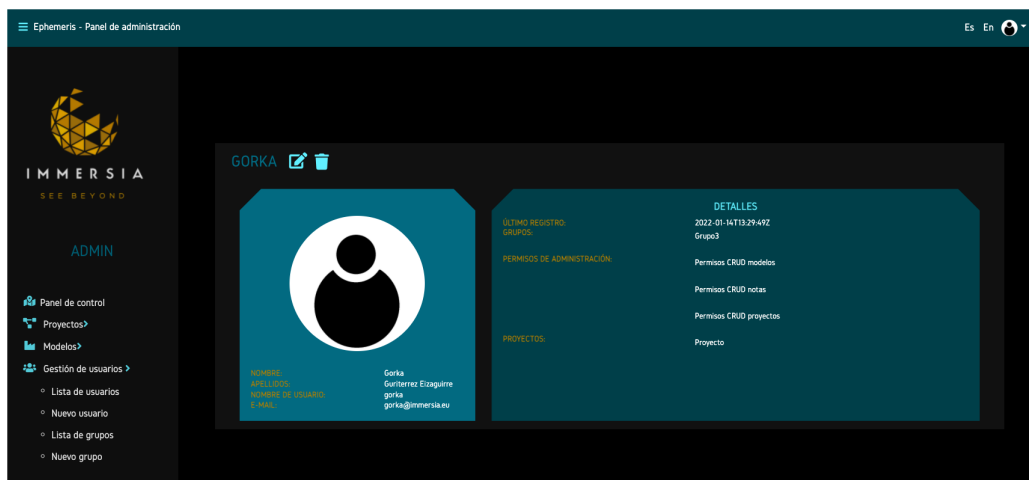


Figura 2.5: Detalle de un Usuario. Tokii Manual de usuario

El usuario, en la ventana correspondiente, podrá visualizar la lista de proyectos y modelos a los que tiene acceso a través de los grupos a los que pertenece, así como sus permisos de administración, en el caso de que tenga alguno.

## 2.3. Aplicación móvil

El acceso a la aplicación móvil se realiza a través de un *login* con usuario y contraseña. Una vez logueado el usuario, la ubicación del dispositivo ha de estar activa para poder usar la app.

Una vez logeado, en la pantalla de búsqueda de ubicación hay tres iconos que indican la correcta conexión del servidor a la app, a internet y a la antena GNSS que garantice establecer una ubicación con una precisión de 2 cm. Si alguno de estos requisitos no se cumple, su respectivo icono estará rojo. Si se cumple, estará en verde.

En el caso de que la precisión de la ubicación no sea de 2 cm, es necesario realizar una calibración en la aplicación. Una vez conectada la antena GNSS, hay que caminar 20 metros en línea recta hacia el poste azul de la pantalla, permanecer en el primer poste azul unos segundos mientras se alinea con el segundo poste. Una vez que el segundo poste se ha puesto verde, se habrá calibrado la aplicación y se cerrará la pantalla de orientación (Figura 2.6).

Una vez que se cumplen las tres condiciones anteriores para el correcto uso de la aplicación, en la pantalla inicial podemos ver los tres botones correspondientes al menú

## 2. EL PROYECTO TOKII



Figura 2.6: Pasos de la Calibración. Tokii Manual de usuario

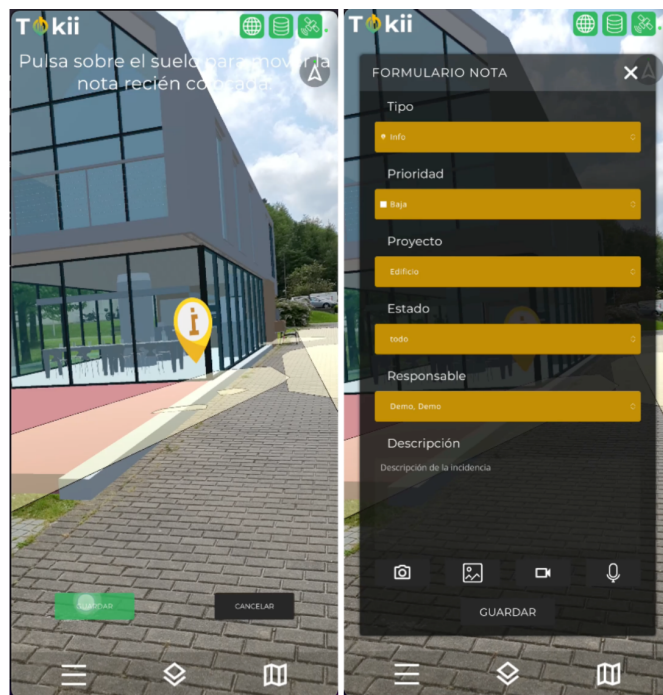
principal, al menú de modelos 3D y a una vista 3D (mapa) donde se puede visualizar la ubicación de las notas y de los modelos.

### 2.3.1. Menú principal

El menú principal dispone de 3 botones:

- Añadir Nota: al clicar el botón se abre el asistente para posicionar una nota, que se situará en el lugar en el que se pulse. Al guardar la nota se abre el formulario de configuración de la nota (Figura 2.7) que contiene diferentes campos (tipo de nota, prioridad, proyecto al que se desea asignar la nota, usuario al que se le desea asignar la nota, descripción y contenido multimedia). Una vez se completa el formulario y se guarda la nota, esta aparecerá geoposicionada en el mundo real y quedará accesible en AR cuando el usuario abra la cámara hacia esa posición. Si no se completa alguno de estos pasos, se cierra el icono generado, la nota desaparecerá y no se almacenará, volviendo a la pantalla inicial de la aplicación.
- Sliders: al clicar el botón se puede modificar el renderizado y la transparencia del modelo que se está visualizando (Figura 2.8).
- Configuración: al clicar el botón se pueden realizar diferentes ajustes como la altura del suelo, el número máximo de actualizaciones, los objetos ocultos hasta posicionamiento, etc.





**Figura 2.7:** Añadir nota en dos pasos (posicionamiento y formulario). *Tokii* Manual de usuario



**Figura 2.8:** Sliders de transparencia y renderizado. *Tokii* Manual de usuario

### 2.3.2. Menú de modelos 3D

El panel de posicionamiento de modelos (menú de modelos 3D del menú principal) permite una visualización geoposicionada cada uno de los modelos, ordenados por proyectos, a los que tiene acceso el usuario.

El panel presenta todos los proyectos a los que el usuario tiene acceso y, por cada proyecto, la lista de modelos asociados al mismo. Pulsando sobre el modelo se despliegan

## 2. EL PROYECTO TOKII

sus capas asociadas. Para posicionar una capa hay que clickar sobre ella y una vez cargada es posible ocultarla/desocultarla pulsando sucesivamente sobre ella. Además, pulsando sobre el nombre del modelo, aparece un panel de edición de su posicionamiento que permite realizar diferentes interacciones:

- **Posición:** permite trasladar el modelo directamente de un punto a otro pulsando en cualquier punto de la pantalla y mover lentamente el objeto una vez posicionado en un punto de izquierda a derecha para realizar un ajuste fino en el posicionamiento deseado.
- **Orientación:** permite rotar el modelo.
- **Altura:** permite subir y bajar el modelo.
- **Escala:** permite ajustar el tamaño del modelo altura y escala del modelo.

Una vez posicionado el modelo también es posible visualizar su información BIM asociada (Figura 2.9).



Figura 2.9: Panel de edición del modelo. Tokii Manual de usuario

### 2.3.3. El mapa

El mapa permite visualizar la ubicación de las notas, que como se ha señalado anteriormente también se puede hacer mediante AR. Es posible interactuar con el mapa y,

al seleccionar una nota, podemos ver toda la información guardada sobre la nota (tipo, proyecto, responsable, etc.) (Figura 2.10).

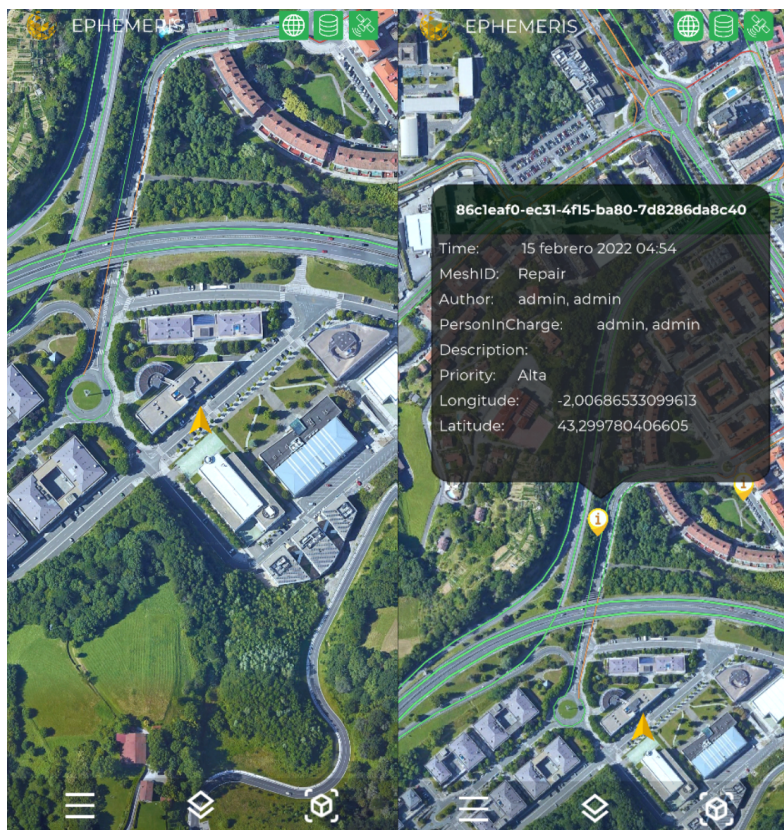


Figura 2.10: Visualización de notas geopositionadas. Tokii Manual de usuario

## 2.4. La API

Una API o interfaz de programación de aplicaciones es un conjunto de definiciones y protocolos que se usa para diseñar e integrar el software de las mismas. En el caso de la plataforma Tokii, la API que se desarrolla es una API REST que se ajusta a los límites de la arquitectura REST<sup>2</sup> y que cuenta con gran capacidad de ajuste. Permite la interacción con los servicios web, de tal forma que podamos comunicar cualquier aplicación o software de la empresa con la base de datos a través de JSON (*JavaScript Object Notation*) y protocolos HTTP.

Así, API REST permite conectar sistemas independientes de forma que cada uno de ellos puede acceder a un recurso dentro cualquier otro y realizar operaciones. Las operaciones más importantes son:

- GET: se utiliza para solicitar que se transmita la información contenida en el recurso.

<sup>2</sup>REST viene de *Representational State Transfer* definido por Roy Fielding en el año 2000 [15]. Fielding establece las restricciones específicas que componen el estilo REST: interfaz uniforme, peticiones sin estado, cacheable, separación de cliente y servidor, sistema de capas y código bajo demanda.

## 2. EL PROYECTO TOKII

---

- PUT: se utiliza para crear o actualizar un recurso en una URL conocida.
- POST: se utiliza para crear un nuevo recurso o enviar datos a un recurso que ya existe para su procesamiento.
- PATCH: permite realizar actualizaciones parciales de un registro. En el cuerpo de la petición se incluirán los cambios a realizar. Puede ser más eficiente en el uso de la red que PUT ya que no envía el recurso completo.
- DELETE se usa para eliminar un recurso existente.
- HEAD: se utiliza para comprobar si existe un recurso

Las operaciones REST siempre se manipulan a partir de la URI, una ruta única personalizada dentro del protocolo HTTP que permite acceder al recurso.

En el caso de la plataforma *Tokii*, la *API REST* permite conectar la aplicación y el panel de control con el servidor, de modo que, cada uno de ellos tenga acceso a los recursos del otro y pueda realizar las operaciones requeridas.

# PLANIFICACIÓN

La planificación es de gran importancia en proyectos a largo plazo. Por este motivo, en IMMERSIA, se utiliza una forma de trabajar metódica para obtener un mejor rendimiento. En este apartado se expondrá tanto la metodología seguida como el software utilizado.

## 3.1. Metodología de trabajo

El desarrollo del proyecto se hace siguiendo metodologías ágiles, que dada su flexibilidad y capacidad de respuesta, permiten adaptar la forma de trabajo a las necesidades de un proyecto. Esto es fundamental para la planificación del mismo. El enfoque de las



**Figura 3.1:** Principios del Manifiesto Ágil

metodologías ágiles es capturado en el “Manifiesto for Agile Software Development”[16] y consiste en priorizar los individuos y sus interacciones por encima de los procesos y de las herramientas, el uso del software mismo frente a la documentación, la cooperación con el cliente frente a la negociación con él y la respuesta al cambio en lugar de seguir un plan establecido [17].

### 3. PLANIFICACIÓN

---

El Manifiesto for Agile consta de doce principios (Figura 3.1[18]) que permiten cumplir estos valores y establecen algunas diferencias entre un desarrollo ágil, que busca una interacción constante, y uno convencional, que es más rígido porque conlleva una serie de fases que hay que cumplir de forma secuencial.

En la actualidad, una de las metodologías ágiles más reconocidas a nivel mundial es *Scrum*, metodología que se ha adoptado para llevar a cabo el proyecto *Tokii* y que se expone a continuación.

#### 3.1.1. *Scrum*

*Scrum* se desarrolla al principio de la década de 1990. *Scrum* es una metodología ágil que utiliza un enfoque incremental fundamentado en la teoría de control empírico de procesos y que está basada en transparencia, inspección y adaptación. La transparencia garantiza la visibilidad en el proceso de las amenazas que pueden afectar el resultado. La inspección ayuda a detectar variaciones indeseables en el proceso; mientras que la adaptación permite realizar los ajustes pertinentes para minimizar el impacto de dichas variaciones [19]. *Scrum* asigna diferentes roles a las diferentes personas que intervienen en un proyecto:

1. Dueño de Producto (*Product Owner*): es la persona responsable del proyecto, que trabaja con el equipo *Scrum* para alcanzar los objetivos del mismo.
2. Equipo de Desarrollo (*Team Developer*): profesionales que realizan las tareas dentro del equipo de trabajo *Scrum*. Pueden ser desarrolladores, testers, analistas, programadores o personas que mantienen la base de datos.
3. *Scrum Master*: es la persona que gestiona el proceso *Scrum*. Modera y ayuda para que el equipo de desarrollo pueda entender cuáles son las necesidades y la metodología de trabajo y ayuda a eliminar los obstáculos que puedan afectar al logro de los objetivos.

El ciclo de vida de *Scrum* abarca diferentes elementos que se resumen en la Figura 3.2[20]:

1. El Dueño de Producto define *Product Backlog*, un documento que contiene la lista completa de necesidades.
2. El Dueño de Producto establece las prioridades y ordena el *Product Backlog* de forma secuencial.
3. *Sprint Planning*: reunión en la que el equipo *Scrum* realiza la planificación del ciclo de trabajo (*Sprint*) con el objetivo de establecer la lista de los objetivos, el plan de acción que se establecen durante ese ciclo (*Sprint Backlog*) y las tareas a desarrollar por el Equipo de Desarrollo.
4. El *Sprint* se pone en marcha con una iteración entre 1 y 4 semanas. En cada *Sprint* intervienen el *Scrum Master* y el Equipo de Desarrollo. Este último es el responsable de dar respuesta a las necesidades que se definen en cada *Sprint*. El *Scrum Master* será la persona encargada de ayudar y de facilitar las cosas para que el Equipo de Desarrollo pueda conseguir este objetivo.
5. El equipo *Scrum* se reúne diariamente para hacer un seguimiento a todos los procesos que se realizan dentro del *Sprint*.

6. Como consecuencia del *Sprint* se obtiene un producto potencialmente entregable que se muestra en la *Sprint Review*, reunión en la que están involucrados tanto el *Scrum Master* como el Equipo de Desarrollo.
7. El ciclo termina con la Retrospectiva del *Sprint*, reunión que busca analizar los resultados del *Sprint* anterior para conocer si ha habido algún problema o fallo en el proceso y proponer mejoras que se puedan aplicar en el siguiente *Sprint*.
8. El proceso se repite hasta completar el *Product Backlog*.

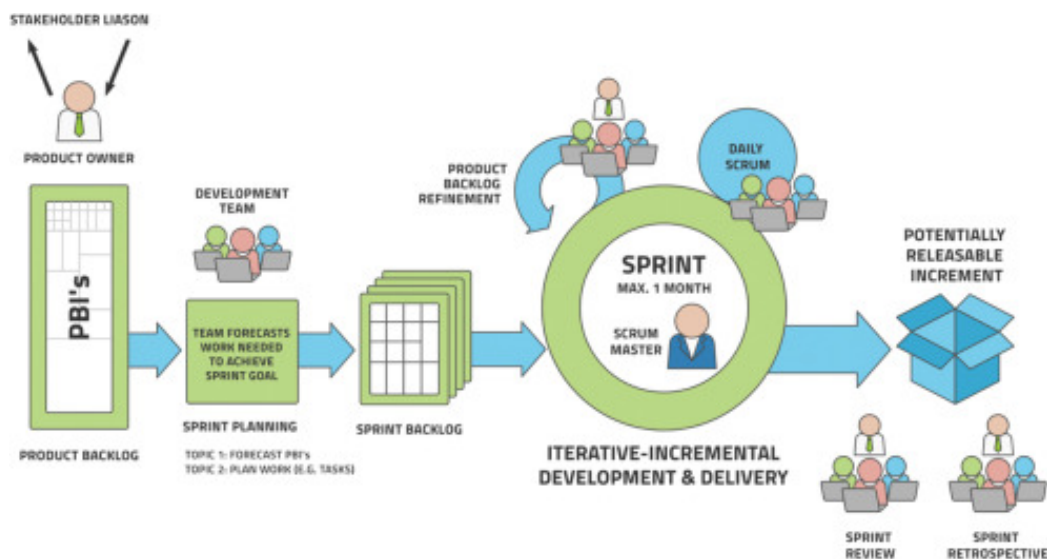


Figura 3.2: Elementos de Scrum

Para gestionar *Scrum* pueden utilizarse diferentes herramientas. El proyecto *Tokii* se ha desarrollado utilizando *Jira*, un software de Atlassian que está diseñado para que todos los miembros de un equipo de software puedan planificar, supervisar y publicar software de calidad asegurada.

En la Figura 3.3 se muestra el tablero de un *Sprint* del proyecto *Tokii*. A cada tarea se le asigna un número de puntos y cada punto equivale a un número de horas de trabajo <sup>1</sup>. Este valor se asigna en el *Sprint Planning* y es orientativo. La persona responsable de la tarea debe realizar un seguimiento sobre su cumplimiento para poder analizar la planificación realizada en la *Sprint Review* <sup>2</sup>.

### 3.1.2. Documentación

Para el desarrollo del proyecto *Tokii*, *Jira* se enlaza con el espacio *Confluence* <sup>3</sup> donde se aloja toda la documentación del proyecto. De esta manera, se elimina la necesidad de utilizar unidades y carpetas compartidas y se accede a la documentación de forma sencilla y directa. Se utiliza para guardar una gran cantidad de documentación como propuestas,

<sup>1</sup>En la empresa IMMERSIA se ha establecido que 1 punto es el equivalente a 4 horas de trabajo

<sup>2</sup>En la empresa IMMERSIA se ha establecido que un *Sprint* es equivalente a 2 semanas de trabajo

<sup>3</sup>*Confluence* es un software de colaboración en equipo. Escrito en *Java* y utilizado principalmente en entornos corporativos, está desarrollado y comercializado por Atlassian.

### 3. PLANIFICACIÓN

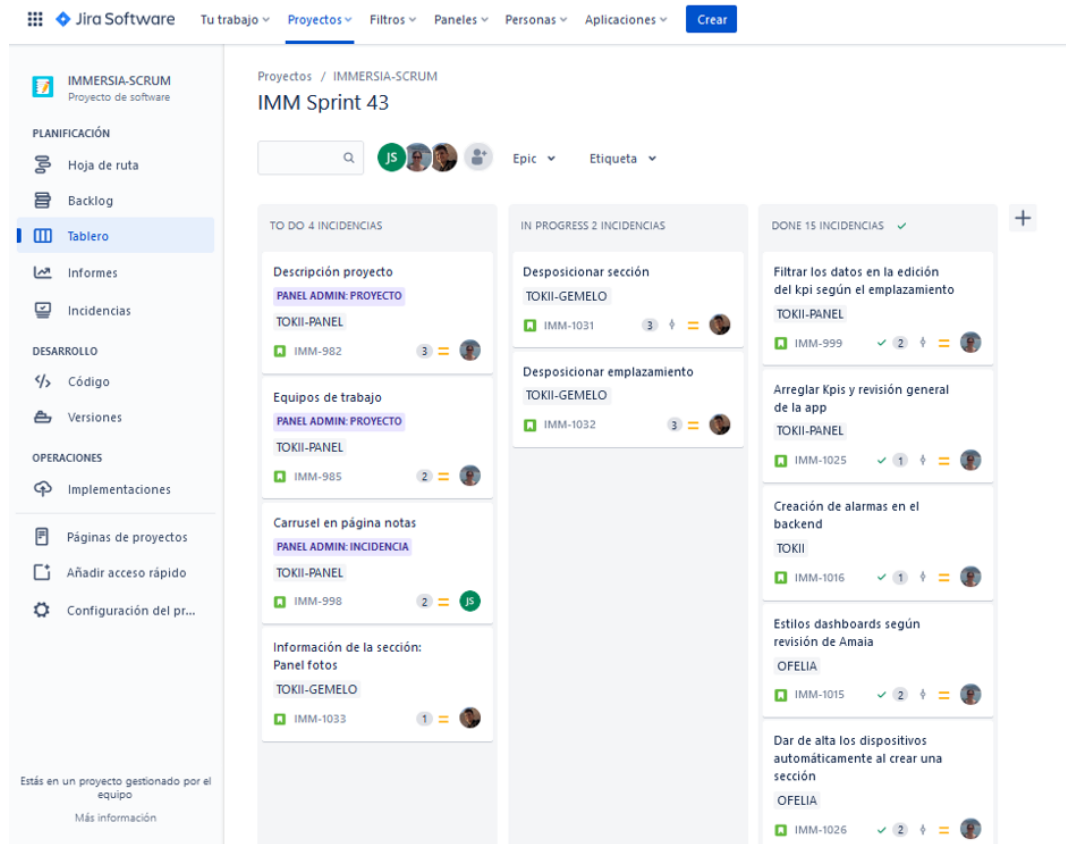


Figura 3.3: Captura de Jira

recopilación de bugs encontrados y cómo se resolvieron, actas de reunión, tutoriales, bases de datos, etc.

#### 3.1.3. Repositorios

La gestión del código del proyecto *Tokii* se ha realizado utilizando *Bitbucket*, un alojamiento para repositorios basados en *Git* (software de control de versiones). Dado que, es un equipo el que realiza el trabajo, se utiliza *Git Flow*, «un modelo alternativo de creación de ramas en *Git* en el que se utilizan ramas de función y varias ramas principales» [21].

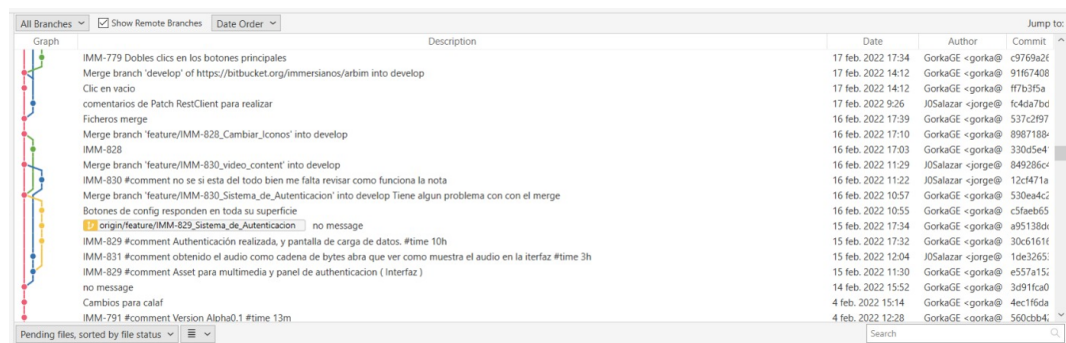


Figura 3.4: Ramas del Bitbucket utilizando Git Flow



En la Figura 3.4 se muestra un ejemplo de un esquema básico de cómo funciona *Git Flow* en el proyecto *Tokii*. La rama roja, llamada *develop*, corresponde a la rama de desarrollo, donde se implementan las funcionalidades en el desarrollo diario; y cada una de las ramas que salen de ésta (con diferentes colores), son las *features*. Cada *features* corresponde a una tarea completa sobre la que trabajan una o varias personas. Las distintas ramas de desarrollo confluyen en la rama *master*, rama principal que se actualiza cada vez que se genera una versión nueva del proyecto.

### 3.2. Entorno de desarrollo

Una vez presentada la metodología de trabajo, se procede a exponer el entorno de desarrollo, es decir, el software que ha utilizado el equipo de desarrollo para la realización de las diferentes tareas asociadas al proyecto.

#### 3.2.1. Unity

*Unity* es una plataforma de desarrollo 3D en tiempo real. Este tipo de plataformas tuvo sus orígenes en el desarrollo de videojuegos y es por este motivo por el que habitualmente se les denomina “motores de juego”. Sin embargo, en los últimos años, su uso se ha extendido a sectores como la arquitectura, la ingeniería, la biotecnología, etc., que han adoptado la tecnología 3D en tiempo real por su gran utilidad.

*Unity* ofrece «capacidades robustas para crear contenido digital interactivo e implementarlo para que se ejecute en varias plataformas, como dispositivos móviles, computadoras y dispositivos montados en la cabeza». Además «permite crear simulaciones en tres dimensiones alimentadas por una gran cantidad de orígenes de datos, por lo que es uno de los entornos más utilizados para crear Gemelos Digitales» [22].

En entornos de VR y AR, *Unity* es el software que más se utiliza y, en el caso de la plataforma *Tokii*, es el software que permite la integración de BIM con AR en la aplicación.

#### 3.2.2. React

*React* es una librería *Javascript* de código abierto desarrollada por Facebook para la creación de interfaces de usuario con el objetivo de animar al desarrollo de aplicaciones de una sola página [23]. Las principales características de *React* son:

1. Mantiene un virtual DOM propio. La lógica de los componentes está escrita en *Javascript*, lo que permite pasar datos de forma sencilla a través de la aplicación y mantener el estado fuera del DOM del navegador.
2. Utiliza una sintaxis llamada JSX que hace el código más legible, aunque es opcional ya que permite otras opciones como HTML.
3. Actualiza y renderiza de manera eficiente los componentes correctos cuando los datos cambian.
4. Los componentes de *React* implementan un método llamado *render()* que recibe datos de entrada y retorna qué mostrar.

### 3. PLANIFICACIÓN

---

5. Permite renderizar desde el servidor usando *Node*, así como potencializar aplicaciones móviles.
6. Permite interactuar con otras bibliotecas y frameworks

En el caso de la plataforma *Tokii*, *React* es la herramienta que permite el funcionamiento del Panel de Control.

#### 3.2.3. *Material Design for Bootstrap (MDB)*

*Material Design* es un lenguaje de diseño desarrollado por Google para unificar el aspecto de la interface. Permite crear un diseño limpio, muy colorido, con efectos dimensionales (iluminación y sombras) y uso de animaciones. La interface creada permite reconocer, a primera vista, lo que es interactivo y lo que no, generando así una web más amigable y mejorando la experiencia de usuario [24].

*Bootstrap* es una biblioteca multiplataforma o conjunto de herramientas de código abierto basado en HTML y CSS, así como en extensiones de *Javascript*, para diseño de sitios y aplicaciones web. Contiene todos los elementos necesarios para el diseño como plantillas, formularios, botones, cuadros, menús de navegación, etc.

*MDB* es un kit de *Material Design* para *Bootstrap* y en el caso de la plataforma *Tokii* es una de las herramientas que se utiliza para el diseño del Panel de Control.

### 3.3. Hitos del proyecto

El proyecto *Tokii* se realiza en diferentes etapas en las que se van superando diferentes hitos. La participación del autor de este TFG en el proyecto engloba los siguientes hitos en los que ha realizado todas las tareas vinculadas a los mismos:

- Hito 1: Creación de contenido multimedia enlazado a las notas.
- Hito 2: Descarga y muestreo de contenido multimedia.
- Hito 3: Apariencia de las notas.
- Hito 4: Modificación gráfica del panel de control.

Los hitos 1, 2 y 3 están vinculados al desarrollo de la aplicación, mientras que el hito 4 hace referencia al desarrollo del panel.

# MEJORAS EN LA APLICACIÓN ANDROID

En este capítulo se presentan las principales actuaciones que se han realizado en la aplicación de android programada en *Unity*.

A continuación se exponen los tres primeros hitos, que tratan sobre la creación y visualización de contenido multimedia enlazado a notas y de la apariencia de las notas.

## 4.1. Hito 1: Creación de contenido multimedia enlazado a las notas

El contenido multimedia aporta información valiosa a la hora de desarrollar un proyecto. Además, si este contenido está geolocalizado, la información es directa e inmediata.

En este sentido, una de las posibilidades que se ha llevado a cabo en la aplicación, ha sido la posibilidad de sacar fotos, vídeos y audios y subirlos a la base de datos, asociando cada elemento multimedia a la nota con la que se relaciona, es decir, geolocalizarlo.

Inicialmente, la única opción multimedia que tenía la aplicación era la de sacar fotos, pero estas fotos, en ningún caso, estaban enlazadas a la nota correspondiente, por lo que se tuvieron que realizar cambios en la aplicación. A continuación se expone cómo se han realizado estos cambios habilitando la posibilidad de enlazar el contenido multimedia creado a la nota correspondiente.

### 4.1.1. Toma de audio y toma vídeo

La toma de fotos estaba realizada con un *asset* llamado *native camara*, el cual llamaba a la cámara del dispositivo para realizar la foto. Este mismo *asset* podía realizar vídeos pero no transformarlos a bytes, lo que es imprescindible para poder enviar los archivos multimedia al servidor. Para obtener los bytes de un vídeo almacenado en una dirección de memoria (*path*) se puede utilizar *System.IO.File* que permite llamar a la función *ReadAllBytes(string path)* que dado un *path* devuelve los bytes escritos en él.

## 4. MEJORAS EN LA APLICACIÓN ANDROID

En el caso de los audios el procedimiento es diferente. En primer lugar, hay que controlar si se tienen los permisos para utilizar el micrófono. Posteriormente, se procede a grabar un clip de audio de *Unity* el cual tiene su propio formato (diferente a los habituales) y un tiempo fijo predeterminado. Para convertir el audio en *Unity* a un formato de audio estándar se debe obtener sus *samples* (amplitud de la onda de sonido en cada momento) y convertirlos a bytes e insertar su correspondiente cabecera. El código base para realizar esta conversión se ha obtenido en un foro de Internet [25] que ha sido modificado de la siguiente manera para que el audio no sea de un tiempo fijo. En primer lugar, se miden los segundos entre la pulsación de comienzo y la de parada y se multiplica el número de segundos obtenidos por la cantidad de *samples* que contiene cada segundo. De esta forma se realiza la conversión parando el bucle en el momento en el que no hay ningún contenido (Figura 4.1).

```
myAudioClip = spectre.audioGrabado;//obtenemos el audio
int ts = spectre.tiemposSamples;//obtenemos el tiempo de grabacion
string pathTemp = Application.persistentDataPath + "/AudioTemp.wav";//path temporal
Save(pathTemp, myAudioClip, ts);//llamamos a guardar el audio con el clip, el path temporal y el tiempo
static void ConvertAudio(FileStream fileStream, AudioClip clip, int ts)
{
    var samples = new float[clip.samples];
    clip.GetData(samples, 0);

    Int16[] intData = new Int16[samples.Length];
    //converting in 2 float[] steps to Int16[], //then Int16[] to Byte[]

    Byte[] bytesData = new Byte[ts * 2];
    //bytesData array is twice the size of
    //dataSource array because a float converted in Int16 is 2 bytes.

    int rescaleFactor = 32767; //to convert float to Int16
    //int f =1;

    for (int i = 0; i < ts; i++) El bucle para dependiendo del tiempo(ts)
    {
        intData[i] = (short)(samples[i] * rescaleFactor);

        Byte[] byteArr = new Byte[2];

        byteArr = BitConverter.GetBytes(intData[i]);

        byteArr.CopyTo(bytesData, i * 2);
    }
    fileStream.Write(bytesData, 0, bytesData.Length);
    //fileStream.Write(bytesData, 0, bytesData.Length);
}
```

Figura 4.1: Código base para convertir el audio en formato *Unity* a un formato de audio estándar

### 4.1.2. Creación de botones

Como se puede observar en la Figura 4.2, en un principio, en la aplicación solo estaba creado el botón del *GameObject* (objeto posicionado en una escena) correspondiente a las fotos. Posteriormente, se crearon todos los demás (audio y vídeo) utilizando la interface

#### 4.1. Hito 1: Creación de contenido multimedia enlazado a las notas

gráfica de *Unity* que permite, a través del inspector, caracterizar todos los componentes del *GameObject*: posición, rotación, funciones, textos, etc. (Figura 4.3) .

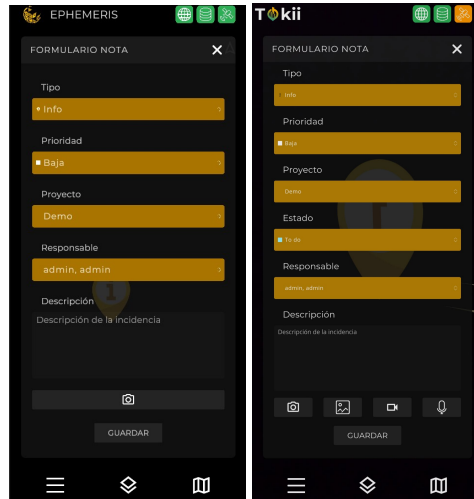


Figura 4.2: Comparación de botones

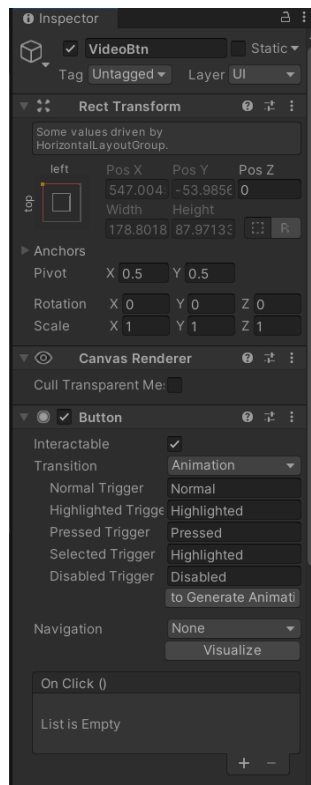


Figura 4.3: Inspector *Unity*

### 4.1.3. Gestión y subida de archivos multimedia a la API

El objetivo es que solo se guarden aquellos archivos multimedia asociados a una nota que se decide guardar.

Para ello se crea una clase controladora *Singelton* que se encarga de controlar los archivos multimedia en el dispositivo con el esquema de la Figura 4.4.

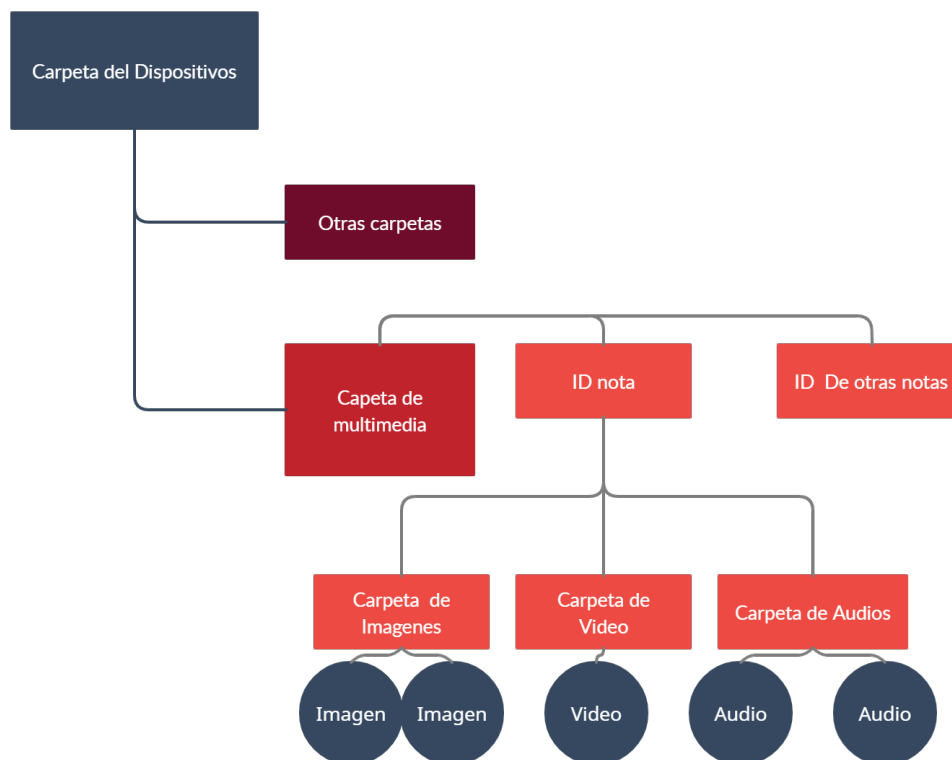


Figura 4.4: Sistema de archivos

La clase controladora incluye una lista de tipo multimedia que es una subclase que contiene el archivo multimedia en bytes, su nombre, su tipo y el directorio donde se va a guardar. Si la nota a la que corresponde el archivo multimedia se guarda, el archivo también se guardará. Si la nota no se guarda, el archivo desaparecerá. La lista se vacía en ambos casos. Este procedimiento permite realizar el control adecuado de archivos temporales.

En el caso de que la nota se haya guardado, todos los archivos multimedia asociados a esa nota se añaden al formulario a mandar desde el controlador del servidor y se envían a la API a través del protocolo POST. Además, se guardan permanentemente en el móvil.

### 4.1.4. Nombrar archivos multimedia

En un principio, la subida del archivo multimedia a la API descrita en el apartado anterior se realizó con un nombre genérico (combinación de caracteres: id incidencia+tipo de multimedia+ fecha, por ejemplo:35c4f614-2fdf-4db4-a39b-5218e5575490\_audio\_08-06-2022T14-42-12.wav). Posteriormente, se crea una ventana que se muestra una vez se realiza la creación, que permite que el usuario ponga el nombre deseado al archivo multimedia una

#### 4.1. Hito 1: Creación de contenido multimedia enlazado a las notas

vez lo haya generado (Figura 4.5) y se sustituye por la fecha. En el caso de que el nombre incluya caracteres no permitidos (dos puntos, interrogación, etc.), estos se eliminan a través de sustitución por expresiones regulares (*Regex.Replace*) (Figura 4.6).

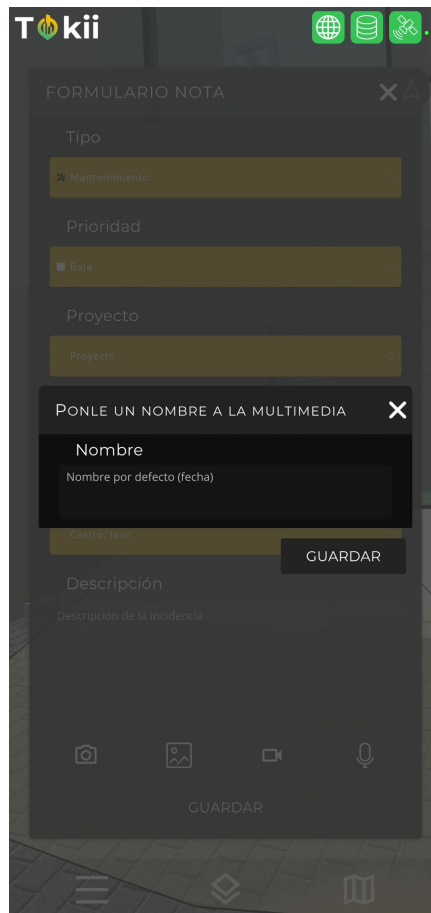


Figura 4.5: Ventana de nombre

```
nombre.Trim();//eliminacion espacio al principio y al final
nombre = Regex.Replace(nombre, @"\s+", "_");//espacios cambiados por _
nombre = Regex.Replace(nombre, @"\)|\|+|\|+|\|+|\|+|\|+|\|+", "");//eliminacion de caracteres no permitidos
nombre = Regex.Replace(nombre, @":+|\|+|\|+|\|+|\|+|\|+|\|+", "");//eliminacion de caracteres no permitidos
textoNombre.text = "";//reiniciamos el cuadro de texto
MediaFiles.Instance.GuradadoTemporal(type, nombre, bytes, laddNota);//lamada al guardado temporal
```

Figura 4.6: Sustitución *Regex.Replace*

##### 4.1.5. Toma de audio, vídeo y foto 2.0

Para la adecuada realización de los modelos 3D en BIM y evitar que existan problemas de calibración, visualización, etc., la grabación de vídeos y la realización de fotos se debe realizar desde la propia aplicación de *Tokii*. En el caso de los vídeos, la grabación se realiza a través de un *asset* llamado *NatCorder* que permite grabar lo que se está mostrando en la pantalla del móvil. Para la toma de fotos se realiza una *Screenshot*.

#### 4. MEJORAS EN LA APLICACIÓN ANDROID

Una vez habilitadas estas opciones, se realizan las interfaces correspondientes a través de *Unity*. Además, con el objetivo homogeneizar todas las interfaces multimedia se realiza una mejora de la interfaz de audio (Figura 4.7).

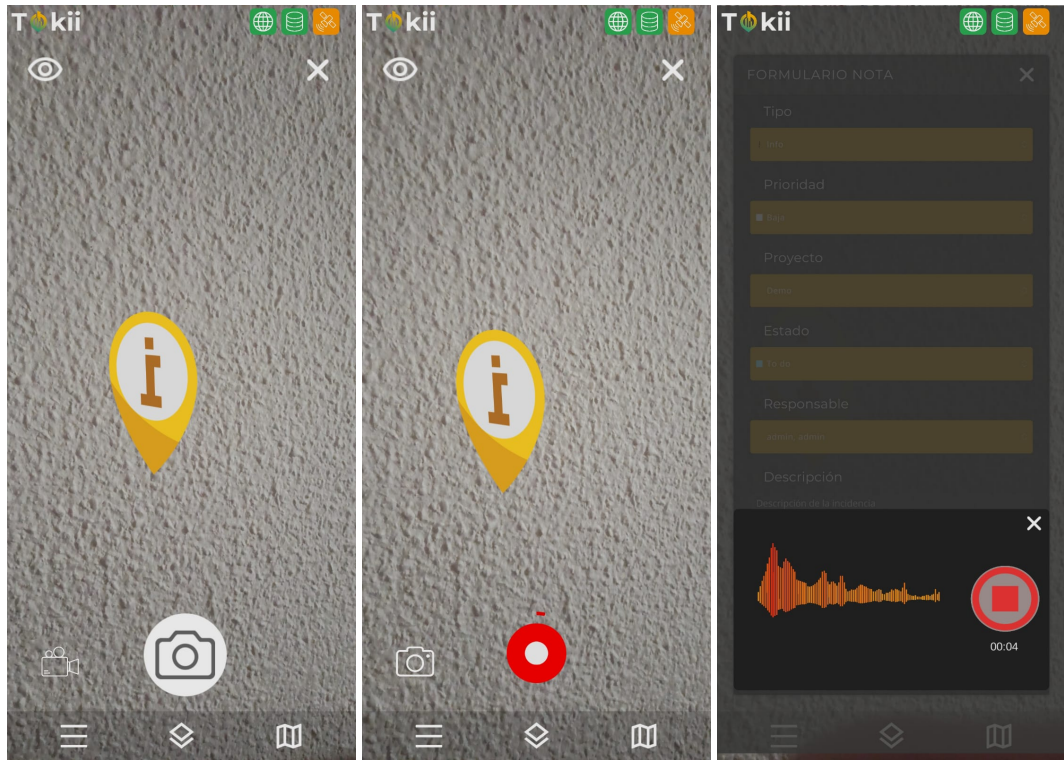


Figura 4.7: Toma de multimedia

##### 4.1.6. *Pacth* y creación de contenido multimedia en la edición de notas

Como se ha señalado anteriormente, hay dos formas de interactuar con las notas: creación y edición. Esto significa que, tanto en la ventana de la creación (“Añadir Nota”) como en la de su edición (“Editar Nota”), se debe permitir al usuario crear archivos multimedia y guardarlos.

En un principio, las distintas formas de captura, tanto de vídeo como de foto, partían de la ventana de “Añadir Nota” que se corresponde con su creación. Esta ventana se ocultaba para realizar la captura y volvía aparecer una vez realizada. Todo este proceso no era posible realizarlo una vez la nota estaba guardada, es decir, no se permitía añadir contenido multimedia en la ventana “Editar Nota”.

Para poder crear y guardar contenido multimedia, tanto cuando se crea una nota como cuando se edita posteriormente, es necesario llamar a las funciones de creación con un booleano que permite controlar desde donde se le llama y así poder ocultar la ventana de la que se parte y volver posteriormente a dicha ventana.

A diferencia de la creación de contenido multimedia desde la ventana de “Añadir Nota”, para la que se utiliza el protocolo *POST*, en el caso de hacerlo desde la pestaña de edición se realiza a través del protocolo *PATCH*. Este último protocolo no funciona adecuadamente



para la librería de *Unity* en la versión que se utiliza para la aplicación *Tokii* por lo que se opta por realizarlo a través de la librería *Resharp*

### 4.2. Hito 2: Descarga y muestreo del contenido multimedia

Una vez creado el contenido multimedia (tanto desde la ventana de “Añadir Nota” como desde la de “Editar Nota”) el siguiente paso es el poder seleccionar y reproducir libremente aquel contenido que se desee cuando se accede a la nota. Esto, además, requiere el control de que se ha realizado la descarga correspondiente.

#### 4.2.1. Pestaña de contenido multimedia

Para mostrar los diferentes contenidos multimedia de una nota, se crea, en la ventana de “Editar Nota”, una pestaña denominada “Multimedia”. Dentro de esta pestaña se encuentran otras tres que separan la multimedia por tipos: imagen, vídeo y audio. Estas pestañas tendrán los correspondientes botones para interactuar con su contenido tal y como se detalla en la Figura 4.8.

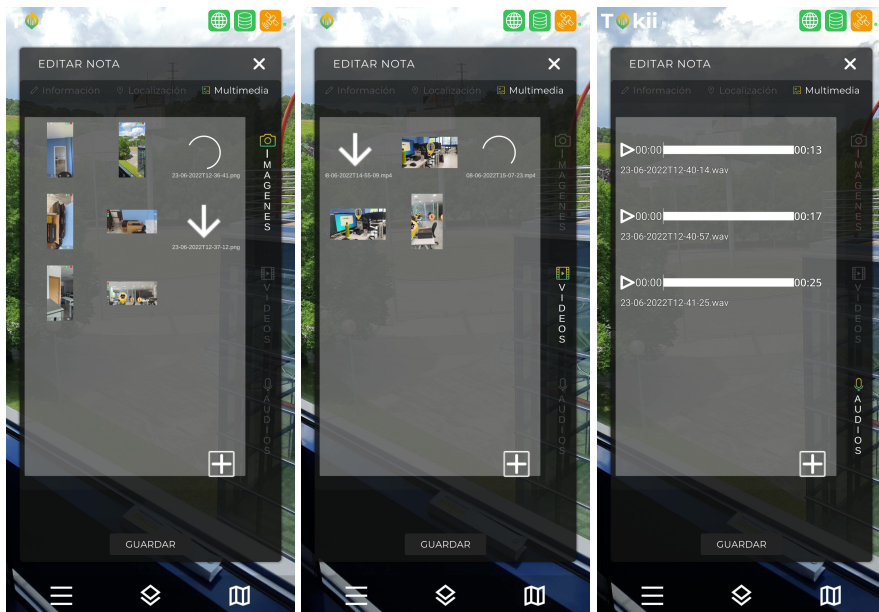


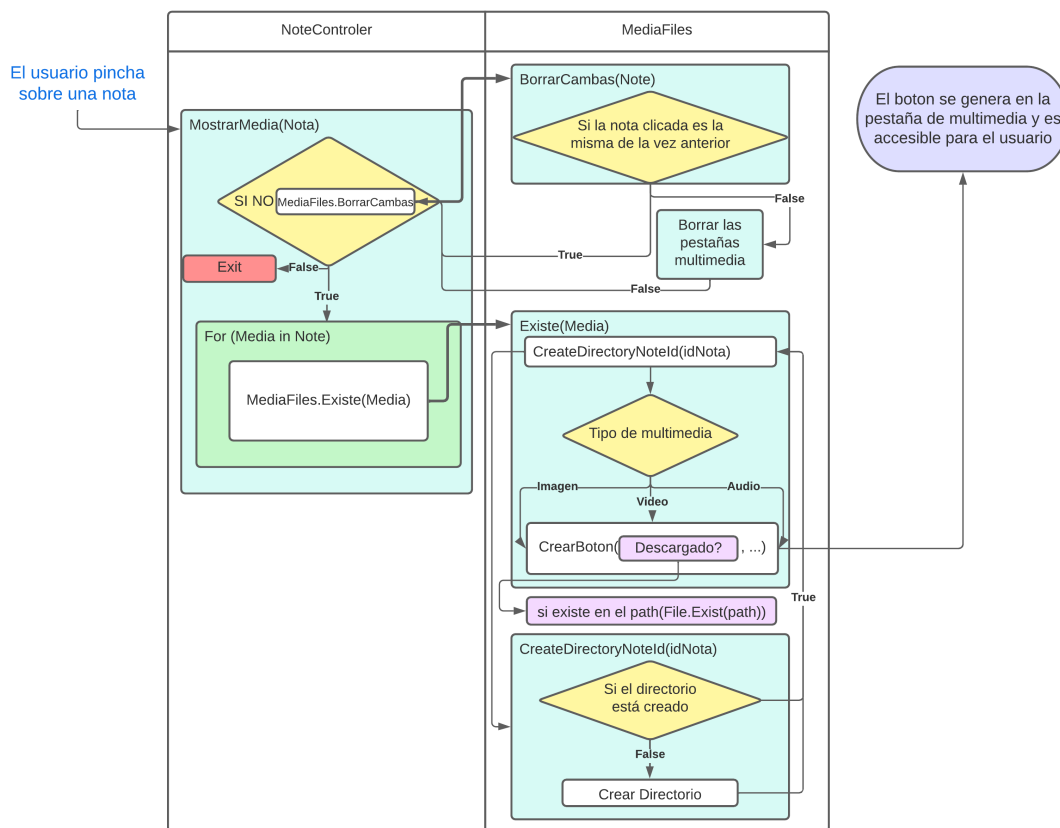
Figura 4.8: Pestaña de multimedia

#### 4.2.2. Botones de contenido multimedia y gestión de descargas

El contenido multimedia llega a través de los *JSONS* que componen las notas. Una vez deserializado el *JSONS*, se obtiene una lista de todo el contenido multimedia de la nota que contiene el ID, el tipo y una URI.

Cada vez que se pincha sobre una nota, se completa la ventana de “Editar Nota” con la información de la misma. Dentro de esa ventana estará ya la pestaña de multimedia por lo que se completará según el Diagrama 4.1.

#### 4. MEJORAS EN LA APLICACIÓN ANDROID



**Diagrama 4.1:** Diagrama de carga de contenido multimedia

Como se puede ver en dicho diagrama, tenemos dos estados: descargado y por descargar. En el caso de que el contenido multimedia esté descargado, se muestra en miniatura y se puede acceder a ver en pantalla completa. Para que estos botones con las miniaturas funcionen, es necesario que cada uno de ellos lleve un script, que será el que se encargue de llamar a la clase controladora del muestreo *ImageVideoControler*.

En caso de que un archivo multimedia no esté descargado, la miniatura es un icono de descarga y más abajo nos muestra el nombre del archivo multimedia. Una vez se interactúa con el botón, se llama a una corrutina de descarga y el icono cambia a un slider circular de carga. Una vez descargado, la dinámica es la misma que en el caso anterior.

#### 4.2.3. Muestra de contenido multimedia

En este apartado solo se explica la muestra de vídeos e imágenes ya solo este tipo de contenido multimedia se muestra en pantalla completa.

Para poder mostrar los vídeos y las imágenes en pantalla completa se ha creado una ventana desde cero con distintos componentes:

1. *RawImage*: se introduce para mostrar la imagen y mantener su calidad de aspecto.
2. Combinación de un *render de vídeo* y un *video player* para poder mostrar el vídeo y controles de vídeo para poder pausar, retroceder y mutear el video.

## 4.2. Hito 2: Descarga y muestreo del contenido multimedia

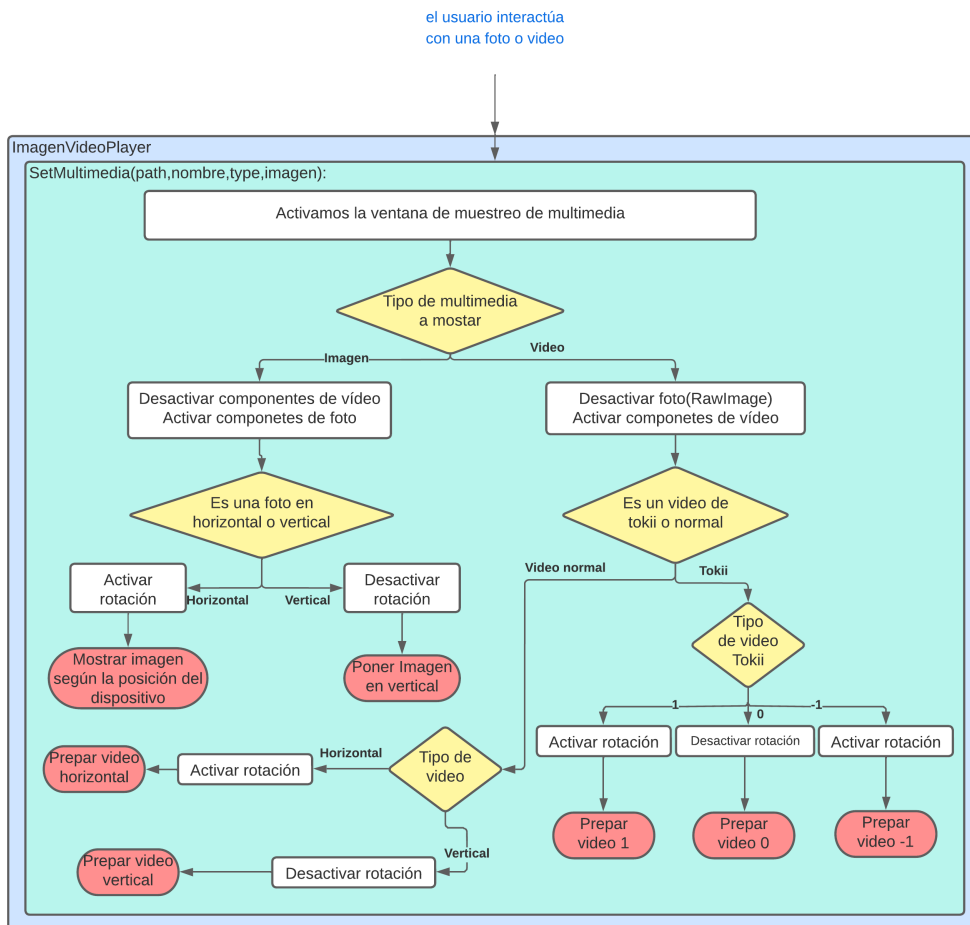


Diagrama 4.2: Diagrama de control de muestra de imagen y vídeo

Además, para controlar el contenido multimedia a mostrar se utiliza *ImagenVideoPlayer*, tal y como se observa en el Diagrama 4.2.

### 4.2.4. Rotación de imágenes

Como se ha podido ver en el diagrama del apartado anterior, se diferencia entre imágenes en vertical y en horizontal. A través de un booleano se impide el giro de las imágenes en vertical, sin embargo, se permite el giro en todas las direcciones de las imágenes en horizontal. Para controlar la rotación del dispositivo se ha programado una corrutina. Adicionalmente, se han incorporado dos botones, el de rotación, que rota la imagen sin necesidad de rotar el móvil, y el de bloqueo, que impide la rotación.

En el botón de rotación, para poder girar la imagen se rota el *gameobjet* donde se encuentra dicha imagen y posteriormente se reestructura (Figura 4.9).

### 4.2.5. Rotación de vídeos

Hay dos tipos de vídeos, los que se graban a través del dispositivo, vídeos comunes, y los creados a través de la aplicación. En un primer momento, se realizó la rotación de vídeos comunes ya que no se había habilitado la posibilidad de grabar vídeos a través

#### 4. MEJORAS EN LA APLICACIÓN ANDROID

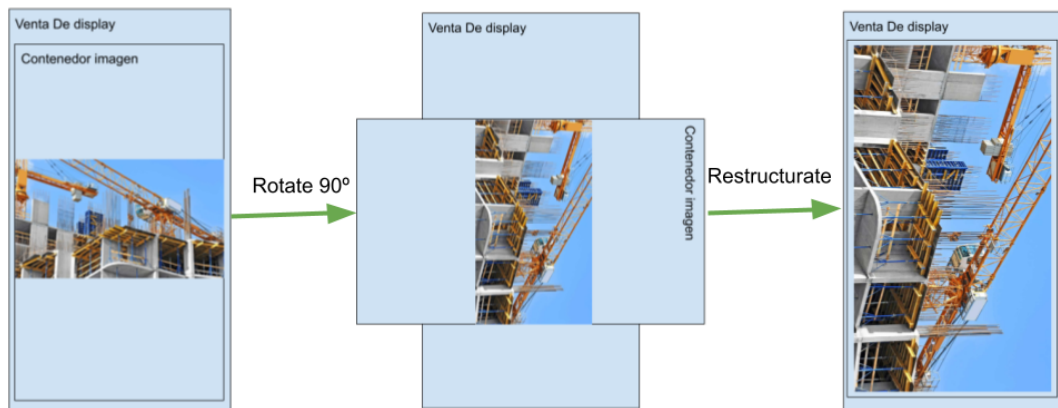


Figura 4.9: Comparación de botones

de la aplicación. En este caso, ajustar la rotación del vídeo implica, además de mover el contenedor del vídeo, ajustar los controles para que el vídeo se mueva.

En el caso de los vídeos grabados a través de la aplicación, los ajustes que se habían estipulado para los vídeos comunes no respondían correctamente ya que, a diferencia de estos últimos no incorporan información sobre la orientación del vídeo. Para reconocer cómo se graba el vídeo dentro de la aplicación, cuando este se guarda, se añade un número al nombre del vídeo (en los caracteres que hacen referencia al tipo de contenido multimedia) que es diferente en función de si se ha grabado en vertical (0), en horizontal izquierda (1) o en horizontal derecha (-1). Esta operación permite controlar la orientación que tiene el vídeo para realizar la rotación correctamente y así realizar los ajustes de los controles correspondientes. Este tipo de ajustes han sido muy tediosos de realizar ya que no es nativo de *Unity* y por lo tanto deben realizarse mediante prueba y error (Figura 4.10).

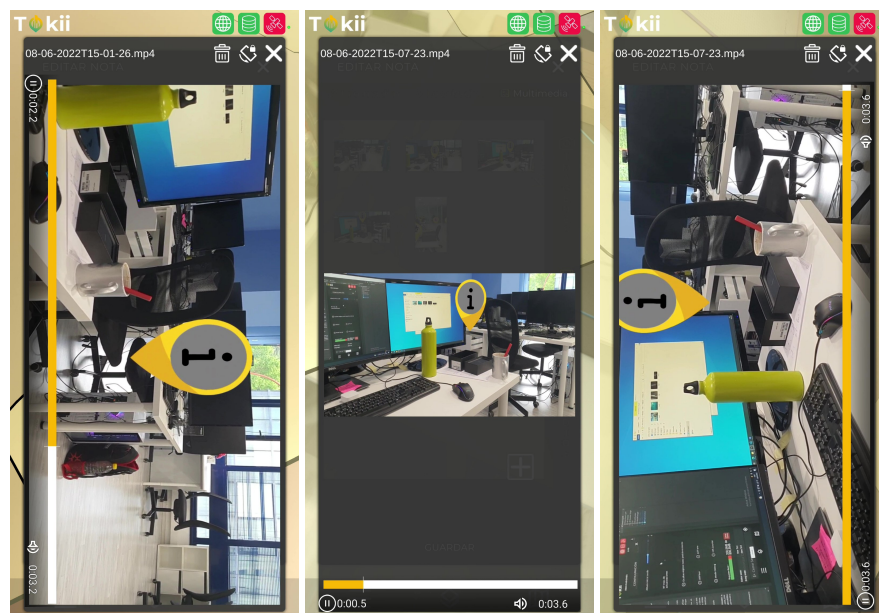


Figura 4.10: Rotación vídeo

### 4.3. Hito 3: Apariencia de las notas

Además del contenido multimedia se han añadido a las notas dos características que se detallan a continuación.

#### 4.3.1. Estado de la Nota

Un usuario debe tener la posibilidad de definir los estados de una nota (por realizar, realizado, peligro, etc.) a través de un nombre y un color que facilite su identificación. Para ello se ha creado una clase estado que se descarga antes que la nota. Completamos el *form* de la nota con los tipos de estados posibles y así el usuario podrá elegir el estado de la nota en el desplegable correspondiente. Una vez realizado este proceso, la nota cambia de color si el estado cambia tal y como se observa en la Figura 4.11



Figura 4.11: Cambio de estado

#### 4.3.2. Type de la Nota

Para permitir que se pueda definir el tipo de nota se sigue el mismo procedimiento que en el caso de la definición de estados, pero en lugar de colores se utilizan iconos que contiene *React* (Figura 4.12). Para encontrar los iconos con el texto del icono se ha generado un *SpriteDataBase* que es una clase que permite al desarrollador almacenar grandes cantidades de datos. En esta clase guardamos el nombre del icono y el propio icono (Figura 4.13).

#### 4. MEJORAS EN LA APLICACIÓN ANDROID



Figura 4.12: Cambio de tipo

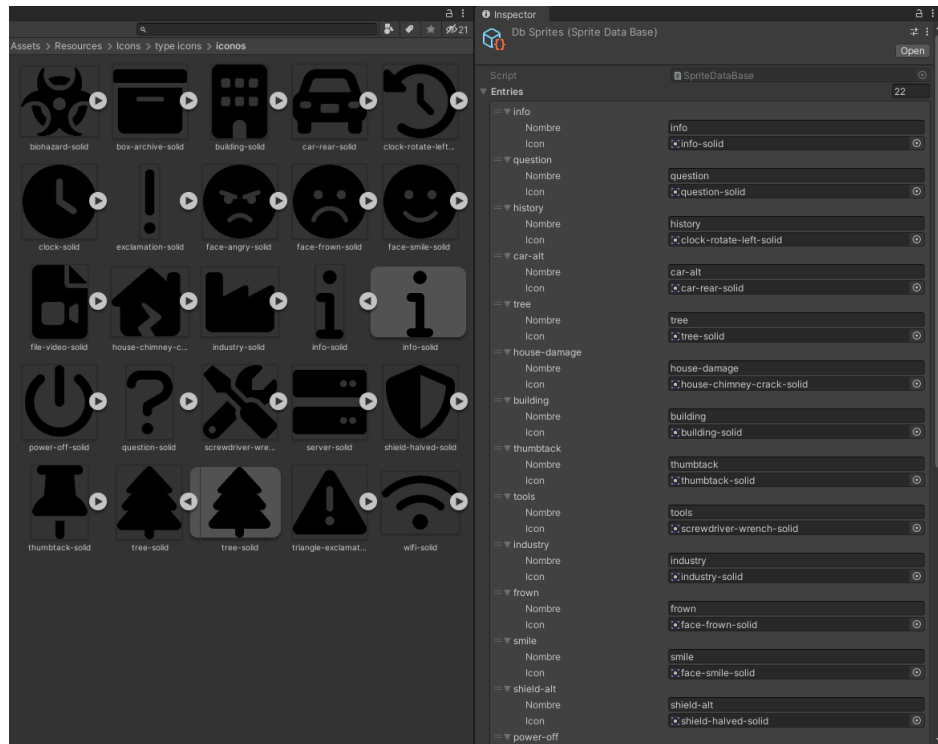


Figura 4.13: SpriteDataBase

# MEJORAS EN EL PANEL DE CONTROL

En este capítulo se describen las diferentes tareas del proyecto *Tokii* que se han llevado a cabo en relación a los hitos correspondientes al desarrollo del Panel de Control.

## 5.1. Hito 4: Modificación gráfica del Panel de Control

La modificación gráfica del panel se ha realizado utilizando *React* y *MDB*, herramientas cuyas características ya se han expuesto en el Capítulo 3 del presente trabajo. Como se puede ver a continuación, los cambios han afectado sobre todo a la estructura gráfica, modificando el CSS de gran parte del Panel de Control.

### 5.1.1. Modificación gráfica del login

En primer lugar se rehace el login del panel, modificando el que previamente había diseñado la empresa de diseño contratada por IMMERSIA. Como se puede observar en la Figura 5.1, la primera imagen se corresponde con el login inicial, el diseño se muestra en el centro y a la derecha la imagen resultante.

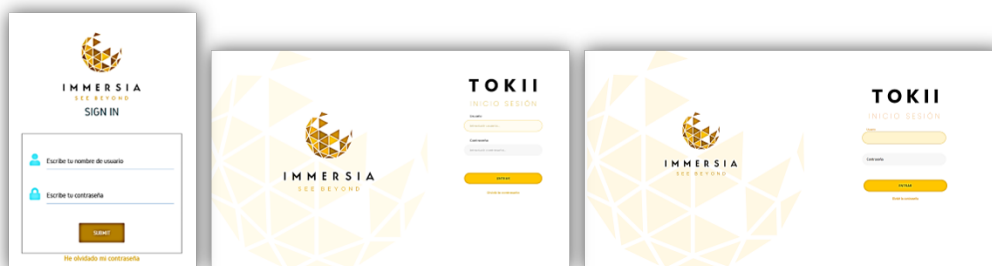


Figura 5.1: Cambio del Login

## 5. MEJORAS EN EL PANEL DE CONTROL

### 5.1.2. Reestructuración del Panel de Control

En este punto se expondrán todas las modificaciones gráficas que se han realizado en el Panel de Control, que como se ha mencionado anteriormente, es el panel central desde donde el administrador y todas las personas que tengan los permisos necesarios van a poder acceder a los diferentes sitios de la página web.

Las secciones en las que se han realizado modificaciones se exponen a continuación.

#### 5.1.2.1. Cabecera

La cabecera se ha diseñado de forma que se compone de tres elementos y ha de estar en todas las páginas del panel. Estos elementos son los siguientes:

- Título del contenido que se presenta en la página que se está viendo. Está situado a la izquierda de la cabecera y puede cambiar en función de si cambia o no el contenido principal al que hace referencia la página.
- El icono. Está situado en el medio y, actualmente, se corresponde con el logo de IMMERSIA, pero, una vez comercializado el producto, se sustituirá por el logo de la empresa cliente.
- Desplegable de los ajustes de usuario. Está situado a la derecha y no es necesario pinchar sobre él, ya que, está programado de tal forma de que si se desliza el ratón por encima, se activa.

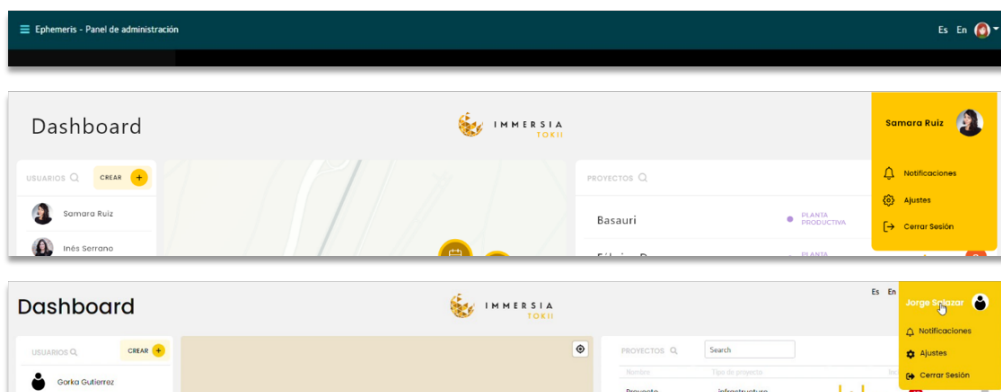


Figura 5.2: Cambio de cabeceras

En la Figura 5.2 se muestra la cabecera inicial, el prototipo y el diseño final una vez efectuados los cambios.

#### 5.1.2.2. Gráficas de notas

En la parte inferior derecha del panel de control se encuentran las gráficas de notas en las que se presenta toda la información relativa a las notas (estado, tipo y notas asociadas a usuarios).

Como se puede observar en la Figura 5.3 se han recreado tres gráficas. Una de ellas se había creado previamente y se han realizado algunas modificaciones, sin embargo, las otras



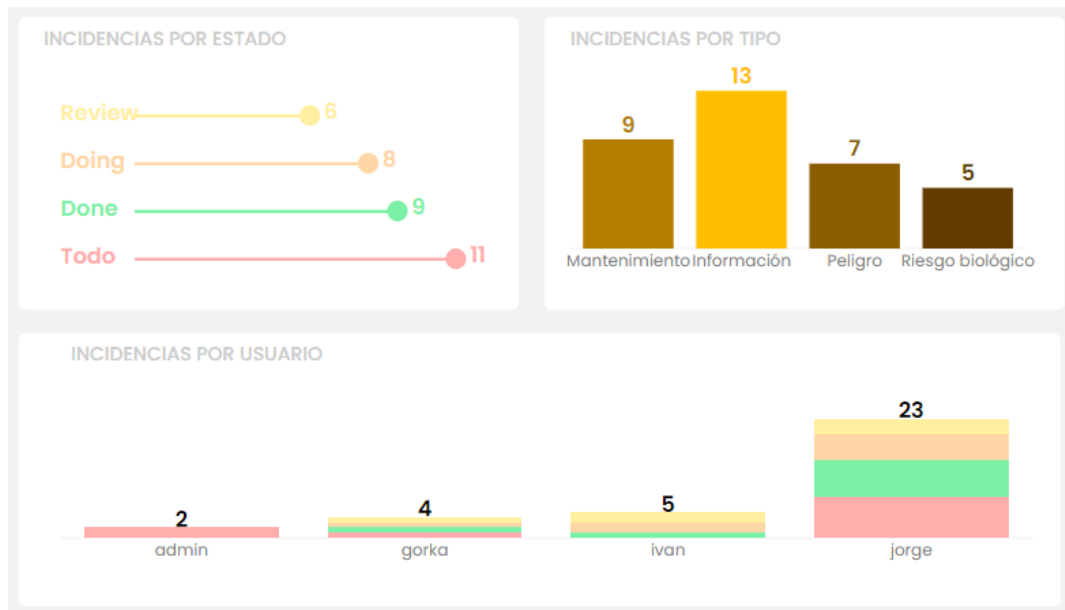


Figura 5.3: Gráficas de incidencias

dos son totalmente nuevas. Para la creación de las gráficas nuevas y la modificación de la existente, se ha utilizado *Ploty*, una librería de código libre muy popular, sobre todo en *Python* y *Javascript*.

La primera gráfica, la horizontal, se corresponde con los estados de las notas (tantos estados como haya definido el administrador) y está compuesta por tres funciones:

- Puntos blancos que se sitúan a la izquierda. Hay tantos puntos como estados y a su izquierda aparece el nombre del estado (Done, To do, etc.) con su color correspondiente. Para que el texto que indica el estado no se solape con la barra horizontal que se ha creado a la derecha, se ha calculado que estos puntos deben colocarse a una distancia equivalente al % 20 del valor máximo de la medida de las barra.
- Barras horizontales muy pequeñas a estilo de líneas que siguen al nombre del estado de la nota. Tiene una longitud variable para indicar el número de notas que hay con ese estado y están coloreadas del color del estado de la nota correspondiente.
- Círculos al final de las barras horizontales. A la derecha del círculo aparece un número que indica cuántas notas hay con ese estado. El círculo y el número están coloreadas del color del estado de la nota correspondiente.

La segunda gráfica está situada a la derecha de la primera y se corresponde con los distintos tipos que se han asociado a las notas (info, repair, danger, etc.). Hay tantos tipos de nota como haya definido el administrador. Esta gráfica se compone de dos funciones:

- Barras verticales. Tienen una altura variable para indicar el número de notas de ese tipo y aparecen coloreadas en amarillo con diferente intensidad: a mayor intensidad (más amarillo), mayor número de notas de ese tipo, a menos intensidad (más marrón), menor número de notas de ese tipo. Para poder calcular este ajuste de color se ha realizado una interpolación lineal.

## 5. MEJORAS EN EL PANEL DE CONTROL

- Puntos blancos con valor “count tipo+1”. Se sitúa el carácter numérico correspondiente al número de notas de un tipo determinado encima de la barra correspondiente a ese tipo.

La tercera gráfica, situada debajo de las anteriores, muestra los números de notas por usuario. Esta gráfica había sido diseñada previamente, pero se realizaron las siguientes modificaciones:

- Homogeneizar el fondo de forma que apareciera en blanco en lugar de cuadrícula.
- Añadir sobre las barras horizontales que muestran el número de notas por usuario, el carácter numérico correspondiente. Esto se hizo de la misma forma que en el caso de la gráfica anterior, utilizando puntos blancos con valor “count tipo+1”.

### 5.1.2.3. Mapa

El mapa está situado en la parte central del panel de control y en el mismo aparecen las notas geoposicionadas. Para que hubiese una concordancia entre el mapa, tal y como aparece en la aplicación, y el mapa, tal y como aparece en el panel de control, se procede a cambiar la imagen de las notas. Así, se establece que una nota es una combinación de tres imágenes: una figura de nota en color naranja, un círculo coloreado con el color de tipo de nota correspondiente y un icono de *React* correspondiente al estado de la nota en el interior del círculo (Figura 5.4).

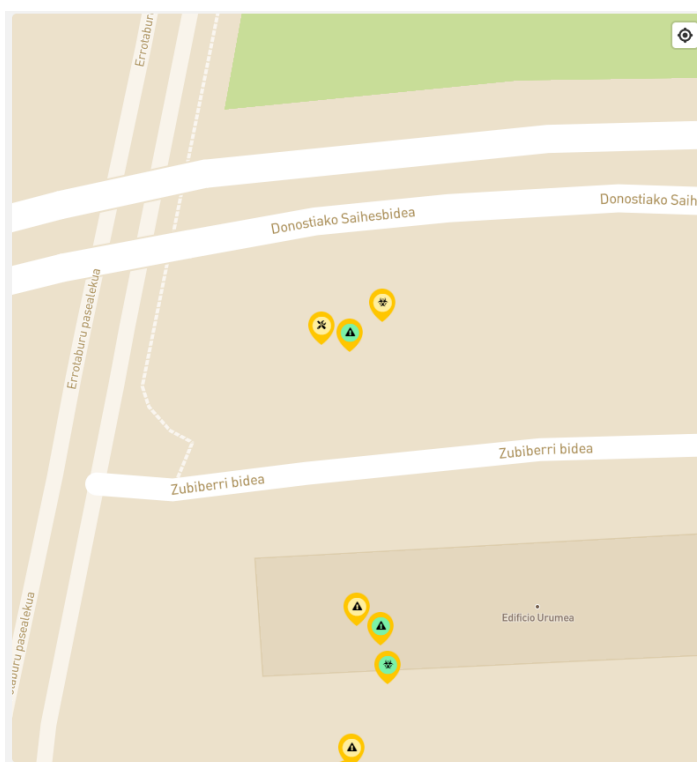


Figura 5.4: Incidencias geoposicionadas en el mapa

#### 5.1.2.4. Tablas de usuarios y proyectos:

Las tablas de usuarios se sitúan en parte de la izquierda del panel de control y las tablas de proyectos a la derecha, encima de las gráficas. La tabla de usuarios muestra todos los usuarios que tienen permiso de acceso a la aplicación y la tabla de proyectos muestra todos los proyectos que ha abierto el administrador del sistema, cada uno de ellos con su nombre, su tipo y dos items que muestran información visual de las notas asociadas.

Este tipo de tablas han sido modificadas respecto a las que había inicialmente a partir de la *datatable* de *MDB* (Figura 5.5).

Name	Position	Office	Age	Start date	Salary
Tiger Nixon	System Architect	Edinburgh	61	2011/04/25	\$320,800
Garrett Winters	Accountant	Tokyo	63	2011/07/25	\$170,750
Ashton Cox	Junior Technical Author	San Francisco	66	2009/01/12	\$86,000
Cedric Kelly	Senior Javascript Developer	Edinburgh	22	2012/03/29	\$433,060
Airi Satou	Accountant	Tokyo	33	2008/11/28	\$162,700
Brielle Williamson	Integration Specialist	New York	61	2012/12/02	\$372,000
Herrod Chandler	Sales Assistant	San Francisco	59	2012/08/06	\$137,500
Rhona Davidson	Integration Specialist	Tokyo	55	2010/10/14	\$327,900
Colleen Hurst	Javascript Developer	San Francisco	39	2009/09/15	\$205,500
Sonya Frost	Software Engineer	Edinburgh	23	2008/12/13	\$103,600

Rows per page: 10 1 - 10 of 14

[SHOW CODE](#) [EDIT IN SANDBOX](#)

Figura 5.5: Ejemplo de *datatable*

A continuación se detallan los cambios realizados en cada una de las tablas y que se pueden ver en la Figura 5.6:

- Se han suprimido los títulos en la tabla correspondiente a usuarios, ya que se considera que la información se refleja correctamente a través de los iconos, y se ha ocultado la función de búsqueda que solo aparecerá si se desliza el ratón por encima de la tabla sustituyendo al título (usuarios).
- En la tabla de proyectos se han introducido dos items que muestran información visual de las notas asociadas a un proyecto: un gráfico y un círculo. En el gráfico

## 5. MEJORAS EN EL PANEL DE CONTROL

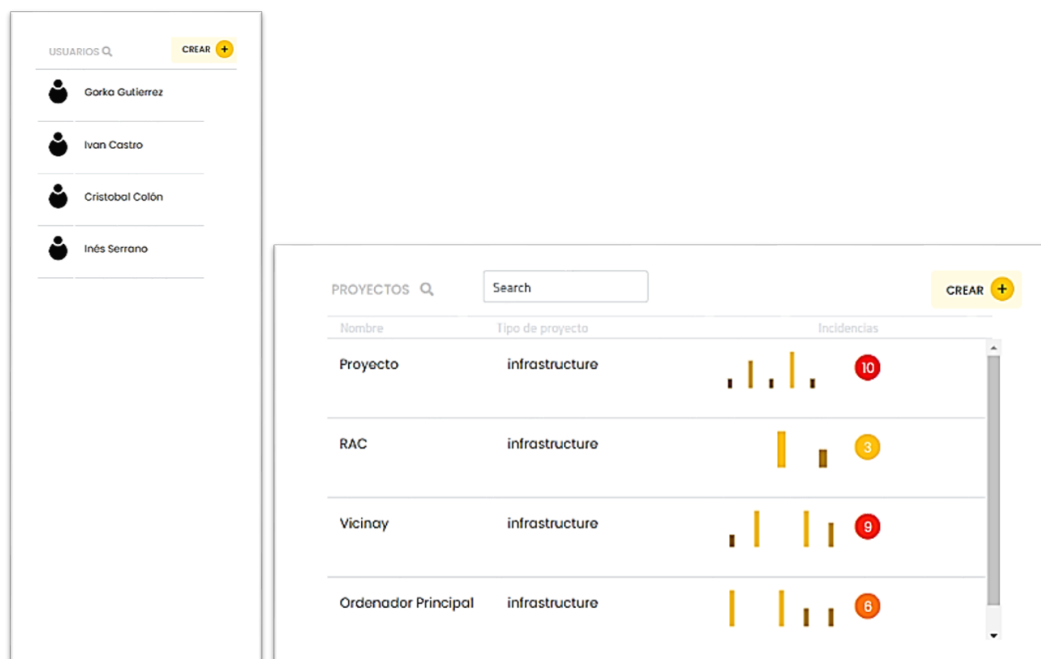


Figura 5.6: Tabla de proyectos y usuarios

aparecen el número de notas que se han realizado en los últimos 6 meses (para ello se ha realizado el algoritmo que puede ver en la Figura 5.7). Aparece una barra por cada mes coloreada en amarillo con diferente intensidad en función del número de notas de dicho mes. El círculo muestra el número de notas abiertas del proyecto y aparece coloreado en rojo con una mayor intensidad para un mayor número de notas. Tanto en el caso de las barras como en el del círculo se ha utilizado interpolación lineal para conseguir la variación en la intensidad de los colores.

```
const month = ["January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"];
//inicializamos una lista con seis ceros donde irá la cuenta de las notas de cada mes desde hace 6 meses
noteInPro = [0, 0, 0, 0, 0, 0]; //ej. mes actual mayo posición 5, diciembre posición 0, febrero posición 2
//siendo la posición 0 es lo correspondiente a hace 6 meses la 1 a hace 5 y así sucesivamente.
//Mismo funcionamiento que la lista anterior pero con los nombres de los meses
meses = ['', '', '', '', '', ''];
//obtenemos el mes actual
const today = new Date().getMonth(); //los meses empiezan en 0
//variable que nos da dos informaciones por una parte los meses que son del año anterior y su relación respecto a la lista
var minimadata = today - 5; // ej. marzo mes 2 -> 2-5=-3 tres meses son del año anterior
props.incidentes.forEach(note => { //for (note in Proyectos)
  //obtenemos las fechas de las notas
  noteday = new Date(note.timestamp).getMonth();
  noteyear = new Date(note.timestamp).getFullYear();
  //pos = posición del mes en la lista (seguimos el ejemplo anterior)
  var pos = noteday - minimadata; //ej. noteday=0 mes enero || 0 --3 =-3 enero esta en la posición 3 de la lista
  //ej.2 noteday=10 noviembre || 10--3=7
  //meses que corresponden a los años anteriores que entran dentro de hace 6 meses
  if (pos > 11) {
    pos = -12/13-12-1 su posición en la lista
    noteyear++; //le añadimos un año porque son meses del año anterior
  }
  if (pos < 6 && noteyear == year) {
    //sumamos 1 al computo de notas del mes en la posición correspondiente de la lista
    noteInPro[pos]++;
    //solo rellenos los meses que tengan una nota o más
    meses[pos] = month[noteday];
  }
});
```

Figura 5.7: Algoritmo de ordenación de meses

## 5.1. Hito 4: Modificación gráfica del Panel de Control

Tras los cambios realizados en el panel de control, en la Figura 5.8 se puede ver la comparativa en relación a la versión inicial del mismo.

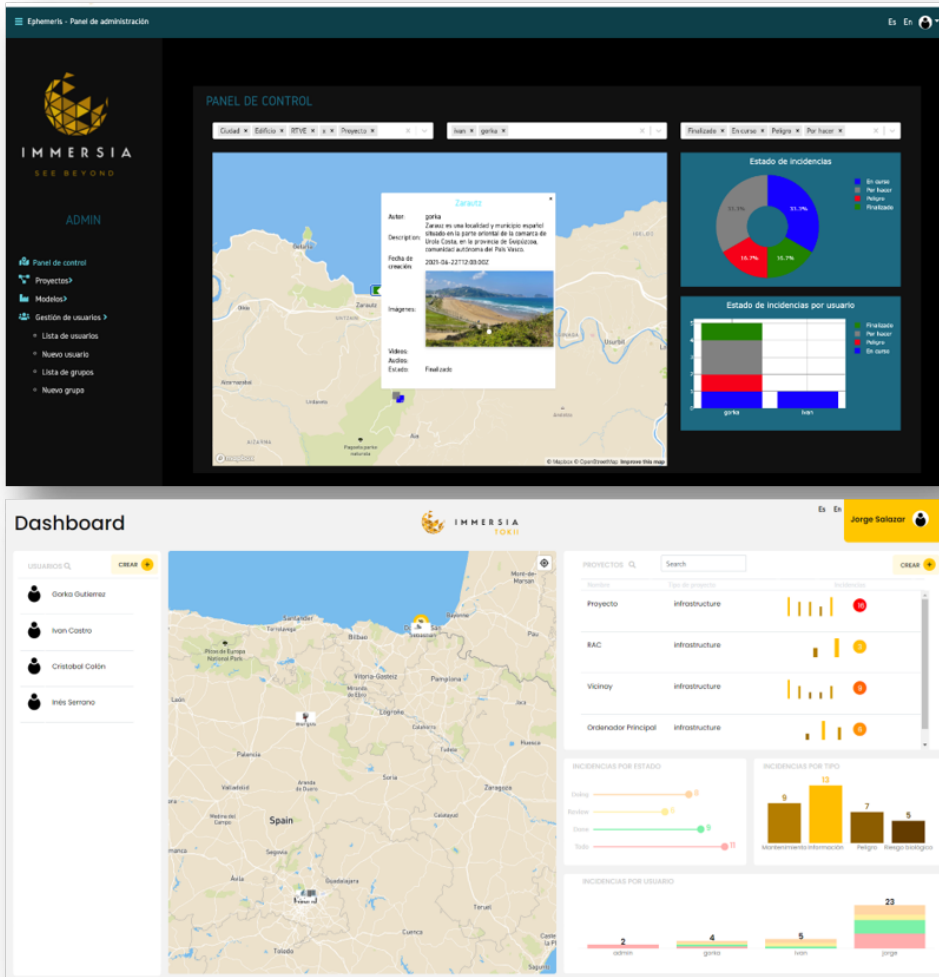


Figura 5.8: Comparativa entre el panel inicial y el final



# CONCLUSIONES, TRABAJO FUTURO Y VALORACIÓN PERSONAL

## 6.1. Conclusiones

BIM hace referencia a una metodología de trabajo colaborativa para la gestión de proyectos de construcción que utiliza modelos 3D digitales abarcando todo el ciclo de vida de un proyecto. Cada una de sus 7 etapas se engloba en una dimensión del método: la idea, el boceto, el modelo 3D, la planificación temporal, la estimación del coste, el análisis de sostenibilidad y la gestión del ciclo de vida. Las plataformas tradicionales de BIM se estancan en las 5 primeras. Entre las más actuales, pocas emplean tecnologías inmersivas para la visualización de los modelos y casi ninguna incluye información geoposicionada. Sin embargo, está comprobado que la integración de AR y GIS en BIM mejora considerablemente la transmisión de la información entre diseño, construcción, operación y fases de mantenimiento y contribuye a implantación progresiva de esta metodología en el sector. En la práctica, el uso de la AR con GIS, permite visualizar los modelos 3D en un entorno real e interactuar con ellos. Así, las plataformas más innovadoras, como *Tokii* unifican la visualización de la información BIM con su información GIS en un entorno AR.

El contenido multimedia aporta información valiosa a la hora de desarrollar un proyecto. Además, si este contenido está geolocalizado, la información es directa e inmediata.

Una de las posibilidades que se ha desarrollado en la aplicación de la plataforma *Tokii* a través de este TFG, ha sido la de sacar fotos, vídeos y audios y subirlos a la base de datos, asociando cada elemento multimedia a la nota con la que se relaciona, es decir, geolocalizándolo. Para que esto sea posible, se han realizado diferentes tareas en la aplicación de android programada en *Unity* que se han presentado en este trabajo y que abarcan todas aquellas que son necesarias para la correcta creación y visualización del contenido multimedia enlazado a las notas.

Adicionalmente, se ha añadido en la aplicación la posibilidad de que el usuario pueda definir el estado y el tipo de una nota geoposiciona, utilizando diferentes colores para el

estado y diferentes iconos para el tipo. De esta manera, una vez definidos el tipo y el estado, estas dos características podrán visualizarse directamente en el icono correspondiente a la nota geoposicionada y permitirán así que el usuario pueda identificar de forma directa información relevante sobre la misma.

En relación al panel de control de la plataforma, los diferentes cambios incorporados se han realizado con el objetivo de mejorar la experiencia del usuario y mantener la concordancia con los elementos comunes con la aplicación. Un panel de control es lo primero que el usuario ve cuando quiere interactuar con la plataforma y tiene que sentirse cómodo y obtener de forma sencilla la información que busca.

### 6.2. Trabajo futuro

En la relación al trabajo futuro en el proyecto *Tokii*, en mi opinión, podrían introducirse algunas posibles mejoras:

1. Mejorar la rotación de vídeo. Si bien se ha conseguido esta posibilidad, considero que la calidad de la rotación debería mejorar y alcanzar niveles similares a los de otras aplicaciones como youtube o twitter. No obstante, considero que esta tarea es bastante compleja.
2. Añadir la posibilidad de eliminar contenidos multimedia. Actualmente es posible eliminar un archivo multimedia en la ventana que se muestra, pero no es posible eliminar varios al mismo tiempo como si se tratara de una galería de un dispositivo móvil

En lo que respecta al panel de control, queda aún mucho trabajo por hacer ya que solo se ha modificado el diseño de la página principal y habría que modificar y homogeneizar el resto.

### 6.3. Valoración personal

La experiencia en la empresa IMMERSIA me ha permitido crecer tanto a nivel profesional como personal. En este sentido, mi participación en el desarrollo de la plataforma *Tokii* ha supuesto un complemento formativo muy importante ya que me ha permitido poner en práctica lo que he aprendido durante mis cuatro años de formación en la universidad. Además, ha contribuido a aumentar mis conocimientos y desarrollar otras competencias y habilidades.

Personalmente, teniendo una base en programación, he pasado de no saber nada sobre Unity y el lenguaje que utiliza, C Sharp, a realizar diferentes tareas vinculadas al desarrollo de la aplicación del proyecto *Tokii*. Además, el haber participado en el diseño de la web me ha permitido conocer herramientas nuevas y desarrollar la creatividad. Es evidente que el aprendizaje adquirido durante este proyecto será de gran utilidad en el futuro para otros proyectos profesionales y para continuar mi formación a través de un Máster.

Trabajar con una metodología ágil ha sido uno de los puntos que me gustaría destacar de toda mi experiencia en el proyecto. Esta metodología me ha permitido adaptar fácilmente mi trabajo a las condiciones del proyecto y tener una comunicación directa y el apoyo de las otras personas que forman parte del equipo de desarrollo. En este sentido, solo tengo



palabras de agradecimiento para mis compañeros y compañeras de IMMERSIA, quienes me han transmitido mucho compañerismo y empatía. Me han tratado como uno más, han compartido conmigo su *expertise* y han resuelto todas las dudas que han surgido en el desarrollo de las tareas.



# Bibliografía

- [1] BuildingSMART Spain. ¿Qué es BIM? Online <https://www.buildingsmart.es/bim/>. 2020. Ver página 3.
- [2] Eastman C. M. et al. An outline of the building description system. *Research Report. Inst. of Physical Planning. Carnegie-Mellon Univ., Pittsburgh, PA*, 50, 1974. Ver página 3.
- [3] Saar C. C. et al. Bim integration in augmented reality model. *International Journal of Technology*, 10(7):1266– 1275, 2019. Ver páginas 4, 6.
- [4] Structuralia. Las 7 dimensiones del BIM y las razones para su dominio. Online <https://blog.structuralia.com>. 2018. Ver página 4.
- [5] Building New Dimensions. ¿Qué tipos de software BIM existen en el mercado? Online <https://www.bimnd.es/tipo-software-bim-en-cada-fase/>. Ver página 5.
- [6] Observatorio Industrial de la Construcción. Informe sobre el Sector de la Construcción 2018. Online <https://www.observatoriodelaconstruccion.com/informes/detalle/informe-sobre-el-sector-de-la-construccion-2018>. 2019. Ver página 5.
- [7] Gosalves J. et al. BIM en 8 puntos. Todo lo que necesitas conocer sobre BIM. Online [https://cbim.mitma.es/recursos\\_cbim/bb\\_gt2\\_personas\\_sg\\_2.1\\_difusion.pdf](https://cbim.mitma.es/recursos_cbim/bb_gt2_personas_sg_2.1_difusion.pdf). 2016. Ver página 5.
- [8] S. Ahmed. Barriers to implementation of building information modeling (bim) to the construction industry: A review. *Journal of Civil Engineering and Construction*, 7(2):107–113, 2018. Ver página 5.
- [9] Carmigniani J. et al. Augmented reality technologies, systems and applications. *Multimed Tool Appl*, 51(1):341–377, 2011. Ver página 5.
- [10] Safikhani S. et al. Immersive virtual reality for extending the potential of building information modeling in architecture, engineering, and construction sector: systematic review. *International Journal of Digital Earth*, 15(17):503–52, 2022. Ver página 6.
- [11] Howari F. M. and Ghrefat H. Geographic information system: spatial data structures, models, and case studies. *Abdel-Mohsen O. Mohamed, Evan K. Paleologos, Fares M. Howari, Pollution Assessment for Sustainable Practices in Applied Sciences and Engineering*, Butterworth-Heinemann(Chap. 4), 2021. Ver página 7.
- [12] Zhu J. and Wu P. BIM/GIS data integration from the perspective of information flows. *Automation in Construction*, 51(1):341–377, 2022. Ver página 7.
- [13] Serrano I. and Ruiz S. BIM Aumentado: el Gemelo Digital en la Construcción. *IMMERSIA White Paper*, 2021. Ver página 9.
- [14] EU Agency for the Space Programme. What is GNSS?. Online <https://www.euspa.europa.eu/european-space/eu-space-programme/what-gnss>. 2021. Ver página 9.
- [15] Fielding R. T. Architectural styles and the design of network-based software architectures. *Doctoral dissertation, University of California, Irvine*, 2000. Ver página 17.

## BIBLIOGRAFÍA

---

- [16] Manifiesto for Agile Software Development. Online <http://agilemanifesto.org/>. 2001. Ver página 19.
- [17] Valencia L. and Herrera E. Del manifiesto ágil sus valores y principios. *Scientia Et Technica*, 34:381–385, 2007. Ver página 19.
- [18] González F. J. Comentando el libro ‘Conceptos ágiles aplicados a distintas áreas de una empresa’, de Roberto Canales. Online <https://www.adictosaltrabajo.com/2020/05/11/>. 2020. Ver página 20.
- [19] Schwaber K. and Sutherland J. The Definitive Guide to Scrum: The Rules of the Game. Online <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>. 2020. Ver página 20.
- [20] Dabrowski P. Project management for a machine learning project. *Machine Learning and Data Science in the Oil and Gas Industry Best Practices, Tools, and Case Studies*, Edited by Patrick Bangert(Chap. 6), 2021. Ver página 20.
- [21] Atlassian. Tutorial GIF Flow. Online <https://www.atlassian.com/es/git/tutorials/comparing-workflows/gitflow-workflow>. Ver página 22.
- [22] Unity. Online <https://unity.com/es/solutions/architecture-engineering-construction>. Ver página 23.
- [23] React Una biblioteca de JavaScript para construir interfaces de usuario. Online <https://es.reactjs.org/>. Ver página 23.
- [24] Google Agencia MDW In Diseño gráfico, Diseño web. Material Design de Google guía en español. Online <https://www.agenciamdw.es/material-design-google-guia-en-espanol/>. 2002. Ver página 24.
- [25] Foro github gist. Unity3D: script to save an AudioClip as a .wav file. Online <https://gist.github.com/darktable/2317063>. 2012. Ver página 26.