

Connect! / Where is my friend?

Descripción de la Arquitectura

Versión 1.0

[Este documento es la plantilla base para elaborar el documento de Descripción de la Arquitectura. Los textos que aparecen entre paréntesis rectos son explicaciones de que debe contener cada sección. Dichos textos se deben seleccionar y sustituir por el contenido que corresponda. Para actualizar la tabla de Contenido, haga clic con el botón derecho del ratón sobre cualquier línea del contenido de la misma y seleccione *Actualizar campos*, en el cuadro que aparece seleccione *Actualizar toda la tabla* y haga clic en el botón Aceptar.]

Historia de revisiones

Fecha	Versión	Descripción	Autor
01/09/2013	1.0	Creación del documento	Dario Britos

Contenido

CONNECT! / WHERE IS MY FRIEND?.....	1
DESCRIPCIÓN DE LA ARQUITECTURA.....	1
VERSIÓN 1.0.....	1
HISTORIA DE REVISIONES.....	1
CONTENIDO.....	2
1. INTRODUCCIÓN.....	4
1. Introducción.....	4
1.1. PROPÓSITO.....	4
1.2. ALCANCE.....	4
1.3. DEFINICIONES, SIGLAS Y ABREVIATURAS.	
WIMF	
WHERE IS MY FRIEND	
4 PLATAFORMAS	
ANDROID, IPHONE, WINDOWS PHONE, WINDOWS 8	
.....	4
1.4. REFERENCIAS.....	4
1.5. VISIÓN GENERAL.....	4
2. VISTA DEL MODELO DE CASOS DE USO.....	5
2. Vista del Modelo de Casos de Uso.....	5
2.1. DIAGRAMA DE CASOS DE USO RELEVANTES A LA ARQUITECTURA.....	5
2.2. CASOS DE USO RELEVANTES A LA ARQUITECTURA.....	5
2.2.1. [Caso de Uso relevante a la Arquitectura 1].....	5
2.2.2. [Caso de Uso relevante a la Arquitectura 2].....	5
3. TRAZABILIDAD DESDE EL MODELO DE CASOS DE USO AL MODELO DE DISEÑO.....	6
3. Trazabilidad desde el Modelo de Casos de Uso al Modelo de Diseño.....	6
3.1.1. [Caso de Uso relevante a la Arquitectura 1].....	6
3.1.2. [Caso de Uso relevante a la Arquitectura 2].....	6
4. VISTA DEL MODELO DE DISEÑO	
.....	7
4. Vista del Modelo de Diseño	
.....	7
4.1. DESCOMPOSICIÓN EN SUBSISTEMAS.....	7
4.1.1. Connect UI.....	8
4.1.2. WIMF UI.....	8
4.1.3. Connect Core.....	8
4.1.4. WIMF Core.....	8
4.1.5. Mobile Data Access.....	8
4.1.6. Connect Services.....	8
4.1.7. WIMF Services.....	8
4.1.8. Security Subsystem.....	8
4.1.9. Server Data Access.....	8
4.2. DISEÑO DE CLASES.....	8
4.2.1. [Clase 1].....	9
4.3. DISEÑO DE CASOS DE USO.....	9
4.3.1. [Diseño del Caso de Uso 1].....	9
5. TRAZABILIDAD DESDE EL MODELO DE DISEÑO AL MODELO DE IMPLEMENTACIÓN.....	10

5. Trazabilidad desde el Modelo de Diseño al Modelo de Implementación.....	10
6. VISTA DEL MODELO DE IMPLEMENTACIÓN.....	11
6. Vista del Modelo de Implementación.....	11
6.1. SUBSISTEMAS.....	11
6.1.1. [Subsistema 1].....	11
6.2. COMPONENTES.....	11
6.2.1. [Componente 1].....	11
6.3. INTERFASES.....	11
6.3.1. [Interfase 1].....	11
7. VISTA DEL MODELO DE DISTRIBUCIÓN.....	12
7. Vista del Modelo de Distribución.....	12
7.1. DIAGRAMA DE DISTRIBUCIÓN.....	12
7.2. NODOS.....	12
7.2.1. [Nodo 1].....	12
7.2.2. [Nodo 2].....	12
7.3. CONEXIONES.....	12
7.3.1. [Conexión 1].....	12
7.3.2. [Conexión 2].....	12

1. Introducción

Este documento tiene como fin, la descripción de la arquitectura de los sistemas Connect! Y Where is my friend?.

1.1. Propósito

El propósito de este documento es proporcionar una apreciación global y comprensible de la arquitectura del sistema usando diferentes puntos de vista para mostrar distintos aspectos del sistema. Capturar y llegar a las decisiones de arquitectura críticas que han sido hechas en el sistema.

1.2. Alcance

Este documento aplica a la estructura base del sistema, interacciones y responsabilidades de sus componentes. Esta principalmente dirigido a diseñadores e implementadores, así como al cliente, ya que en este proyecto se hizo énfasis en la descripción de la arquitectura.

1.3. Definiciones, siglas y abreviaturas.

WIMF

Where is my friend

4 Plataformas

Android, Iphone, Windows Phone, Windows 8

1.4. Referencias

[1] Mosync <http://www.mosync.com/>

1.5. Visión general

[Esta sección describe que contiene el resto del documento **Descripción de la Arquitectura** y explica cómo se organiza este documento.]

2. Vista del Modelo de Casos de Uso

2.1. Diagrama de Casos de Uso relevantes a la Arquitectura

[En esta sección se muestra un diagrama de Casos de Uso incluyendo solo los casos de uso relevantes a la Arquitectura del sistema.]

2.2. Casos de Uso relevantes a la Arquitectura

[En esta sección se detallan los Casos de Uso y actores más significantes para definir la Arquitectura del sistema. Estos casos de uso son:

- los que ayudan a mitigar los riesgos más serios
- los más importantes para los usuarios y clientes
- los que ayudan a cubrir toda la funcionalidad importante para que nada quede en sombras]

2.2.1. [Caso de Uso relevante a la Arquitectura 1]

[Descripción del Caso de Uso]

2.2.2. [Caso de Uso relevante a la Arquitectura 2]

[Descripción del Caso de Uso]

...

3. Trazabilidad desde el Modelo de Casos de Uso al Modelo de Diseño

[Describir la trazabilidad entre el Modelo de Casos de Uso y el Modelo de Diseño. Identificando los objetos y subsistemas de diseño que intervienen en el caso de uso y sus relaciones (Diagrama de paquetes).]

3.1.1. [Caso de Uso relevante a la Arquitectura 1]

[Diagrama de paquetes para el Caso de Uso.]

3.1.2. [Caso de Uso relevante a la Arquitectura 2]

4. Vista del Modelo de Diseño

4.1. Descomposición en Subsistemas

4.1.1. Connect UI

Es el front-end de la aplicación Connect!, el cual tiene 4 variantes, uno para cada plataforma (Android, iPhone, Windows Phone, Windows 8). Se conecta única y directamente con el núcleo Connect.

4.1.2. WIMF UI

Es el front-end de la aplicación Where is my Friend?, el cual tiene 4 variantes, uno para cada plataforma (Android, iPhone, Windows Phone, Windows 8). Se conecta única y directamente con el núcleo WIMF.

4.1.3. Connect Core

Primer componente del core común a las 4 plataformas. Contiene toda la lógica de negocio de la aplicación Connect. Esta implementado puramente en C++ y debe ser compilable para las 4 plataformas. Este núcleo estará implementado con el framework Mosync [1]. Entre sus características se pueden destacar: La interacción con las Api's de las redes sociales, la obtención de datos de intercambio entre móviles (cámara + QR).

4.1.4. WIMF Core

Segunda componente del núcleo común a las 4 plataformas. Contiene toda la lógica de negocio de la aplicación Where is my friend?. Esta implementado puramente en C++ y debe ser compilable para las 4 plataformas. Este núcleo estará implementado con el framework Mosync [1].

Entre sus características se pueden destacar: Utilización del GPS del dispositivo, despliegue de ubicaciones en mapas.

4.1.5. Mobile Data Access

Tercer componente del núcleo común. Pequeño subsistema que se encarga del almacenamiento de datos local a cada móvil.

4.1.6. Connect Services

Subsistema que provee con los servicios necesarios para el funcionamiento de la aplicación Connect!. Dichos servicios estarán publicados mediante la tecnología REST, con acceso seguro https. Estará implementado en C# y expuesto como Web Service.

4.1.7. WIMF Services

Subsistema que provee con los servicios necesarios para el funcionamiento de la aplicación Where is my friend?. Dichos servicios estarán publicados mediante la tecnología REST, con acceso seguro https. Estará implementado en C# y expuesto como Web Service.

4.1.8. Security Subsystem

Es un subsistema que encapsula todo el trabajo de seguridad del servidor. Se separó en un subsistema porque es común a ambas aplicaciones.

4.1.9. Server Data Access

Subsistema que provee las funcionalidades necesarias para el almacenamiento y la recuperación de datos.

4.2. Diseño de Clases

[Se indican las Clases de Diseño claves del Sistema.]

4.2.1. [Clase 1]

[Breve descripción de la Clase, incluyendo principales funcionalidades que brinda, esto se encuentra con mayor detalle en el Modelo de Diseño.]

4.3. Diseño de Casos de Uso

4.3.1. [Diseño del Caso de Uso 1]

[Describe las clases del Diseño y los subsistemas que corresponden a la Arquitectura del Diseño del Caso de Uso 1.]

5. Trazabilidad desde el Modelo de Diseño al Modelo de Implementación

[Indicar la trazabilidad entre las entidades de diseño del Modelo de Diseño y los componentes del Modelo de implementación.

Indicar para cada entidad de diseño qué elemento en el ambiente de implementación le corresponde, por ejemplo para cada subsistema el módulo que lo implementa; para una funcionalidad importante cual es el conjunto de objetos que la implementan, a qué módulo pertenece.]

6. Vista del Modelo de Implementación

[En esta sección se describe la estructura general del modelo de implementación y la descomposición del software en módulos. Utilice los subtítulos a continuación para esta descripción.]

6.1. Subsistemas

6.1.1. [Subsistema 1]

[Se detalla brevemente y se muestra la correspondencia con el Modelo de Diseño del Subsistema 1 relevante a la Arquitectura.]

6.2. Componentes

6.2.1. [Componente 1]

[Se especifica la correspondencia con el Modelo de Diseño del Componente 1 implementado relevante a la Arquitectura. También se debe especificar cómo están organizados los componentes de acuerdo a la estructura y mecanismos de modularización disponibles en el ambiente de implementación y en el(los) lenguaje(s) de programación, y cuales son las dependencias entre los distintos componentes. Pueden existir dependencias de compilación entre componentes, denotando cuales componentes son requeridos para la compilación de un componente específico.]

6.3. Interfases

6.3.1. [Interfase 1]

[Se detalla brevemente y se especifica la correspondencia con el Modelo de Diseño de la Interfase 1 relevante a la Arquitectura.]

7. Vista del Modelo de Distribución

7.1. Diagrama de Distribución

[El modelo de distribución describe la distribución física del sistema en términos de como se distribuye la funcionalidad entre los nodos computacionales.

Realizar uno o varios diagramas de los nodos relevantes a la arquitectura del sistema, como son las conexiones entre ellos y como se distribuye la funcionalidad en ellos.]

7.2. Nodos

[En esta sección se da más detalle de los nodos relevantes a la arquitectura del sistema.]

7.2.1. [Nodo 1]

[Describir las características y capacidades del nodo. También se debe indicar el subsistema de Diseño asociado al nodo, de esta forma se muestra la correspondencia entre la Arquitectura del Software y la Arquitectura del Hardware.]

7.2.2. [Nodo 2]

...

7.3. Conexiones

7.3.1. [Conexión 1]

[Describir las características de la Conexión 1 entre nodos del sistema, como ser tipo de conexión, ancho de banda, disponibilidad, etc.]

7.3.2. [Conexión 2]

...