**Instituto Tecnológico de Costa Rica**

Área de Ingeniería en Computadores

CE-4101 Especificación y Diseño de Software



Proyecto Digi-Tutor

Descripción Arquitectura/Descripción

Detalla de Diseño (AD/DDD)

Elaborado por:

* Arturo Chinchilla 2013009344
* Fabián Solano Rodríguez 2013116291
* Juan Navarro Camacho 201236227
* Malcolm Davis Steele 201271325
* Santiago Gamboa Ramírez 2014092362

Profesor:

Ing. Daniel Madriz Huertas

Cartago

Segundo Semestre 2017

[**Introducción.**](#_rzoz98qme2yd) **3**

[Control de documento.](#_fpw507ctghgu) 3

[Tabla de definiciones y acrónimos.](#_hywyemeonh2b) 4

[Visión general del documento.](#_owr8v8q5yhj) 5

[**Descripción de Arquitectura.**](#_szvt80tgfvj5) **7**

[Diagrama de Contexto con tabla de descripción.](#_8gj9p2t1yb9k) 7

[Modelo de estructura modular](#_k2h04geiyp4w) 9

[Descripción de capas del sistema Digi-Tutor](#_72i8jcorrkhy) 10

[**Requerimientos Funcionales**](#_vsx1xfsakop) **10**

[Diagrama de Casos de Uso.](#_rviudh6e6uu6) 10

[Diagramas de Secuencia de clases.](#_vd2bltdgcr12) 10

[Diagrama de Clases de Diseño.](#_julutgfbuez1) 10

[Descripción de patrones de diseño utilizados.](#_jhahs624ean8) 10

[**Diseño de Datos.**](#_133v5tyigyqd) **12**

[Diagrama de Base de Datos](#_xlxs8xomx8zu) 12

[**ANEXOS**](#_dntiensmxsi9) **13**

## Introducción.

### Control de documento.

Cuadro 1: Historial de revisión

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Fecha de Revisión | Autor | Descripción del Cambio |
| 1.0 | 27/11/2017 | JAMS | Se crea el documento |
| 2.0 | 2/12/2017 | JAMS | Se agregaron diferentes diagramas |

Cuadro 2: Identificación de los cambios y de la versión vigente de los Documentos

|  |  |
| --- | --- |
| Versión Inicial | 1.0 |
| Versión Actual | 2.0 |
| Fecha de última versión | 3/12/2017 |

Cuadro 3: Control de documentos externos

|  |  |
| --- | --- |
| Documento | Ubicación |
| 1.Proyecto-Digi-Tutor\_v1.4.pdf | tecDigital |
| Company Profile WorkTalent v2.3.pdf | Correo |
| ERS-Simplificando v2.6.pdf | Correo |

### Tabla de definiciones y acrónimos.

|  |  |
| --- | --- |
| **Término** | **Definición** |
| Administrador | Rol de un usuario, creado por Digi-Tutor para modificar diferentes valores del sistema. |
| Dashboard | Es una herramienta para mostrar diferente contenido a los tutores y administradores. |
| Deshabilitado | No permite ser utilizado en el sistema. |
| Habilitado | Que se encuentra disponible para realizar acciones dentro del sistema |
| Login | Nombre de usuario de autentificación necesario para acceder al sistema. |
| Tutoría | Publicación hecha por un tutor para crear un “evento” al cual otros tutores se inscriben |
| Contenido Académico | Publicación hecha por un tutor para compartir contenido académico. |
| Almacenamiento Blob | Servicio de almacenamiento en la nube proporcionado por Azure |
| Perfil | Conjunto de datos de un usuario en el sistema |
| Apoyo | Voto a favor de la habilidad de un tutor |
| Habilidad | Habilidad que el usuario (tutor) cree que domina, seleccionada desde el sistema |
| Reputación | Puntuación recibida por el Tutor con respecto a la cantidad de likes y dislikes recibidos. |
| Reporte | Salida del sistema para el administrador. |
| Twitter | Es un servicio de microblogging |
| User story | Consiste en una representación de un requisito escrito en una o dos frases utilizando el lenguaje común del usuario. |
| Usuario | Persona física que utiliza el sistema para sus intereses |
| Web service | Tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones |

|  |  |
| --- | --- |
| **Acrónimos** | **Significado** |
| AES | Advanced Encryption Standard |
| ASP | Active Server Pages |
| GIF | Graphics Interchange Format |
| JPG | Joint Photographic Experts Group |
| HTML | HyperText Markup Language |
| PNG | Portable Network Graphics |
| REST | Representational State Transfer |
| SQL | Standard Query Language |
| SRS | Software Requirements Specification |
| DSS | Especificación de Requerimientos de Software |

### Visión general del documento.

El presente documento describe el diseño detallado de software del sistema DigiTutor. En el mismo se incluyen diagramas que representan la organización, estructura e interacción del código con diferentes capas internas y servicios externos.

El presente documento describe el diseño detallado de software del sistema Digi-Tutor, el objetivo es crear una comunidad para compartir contenido y encontrar talento humano por parte de los colaboradores de la empresa Work Talent quienes analizan la opinión de la comunidad durante los procesos de selección de talento, utilizando su propio algoritmo. La plataforma sirve también como medio de comunicación entre tutores, donde cada uno comparte su contenido y puede ser accedido por tutores que se han suscrito a dicho material. Además existen interacciones como puntajes y comentarios entre los tutores con el fin de cuantificar las habilidades de cada tutor.

Este documento de Diseño Detallado del Software del sistema Digi-Tutor es la base de la codificación del mismo. Describe los siguientes tópicos: un diagrama de contexto con su tabla de descripción, un modelo de estructura modular con su respectiva descripción de capas, diagramas de casos de uso, diagramas de secuencia de caso de uso para escenarios principales, un diagrama detallado de clases de diseño, descripción de patrones de diseño utilizados y el diagrama de bases de datos.

## Descripción de Arquitectura.

### Diagrama de Contexto con tabla de descripción.

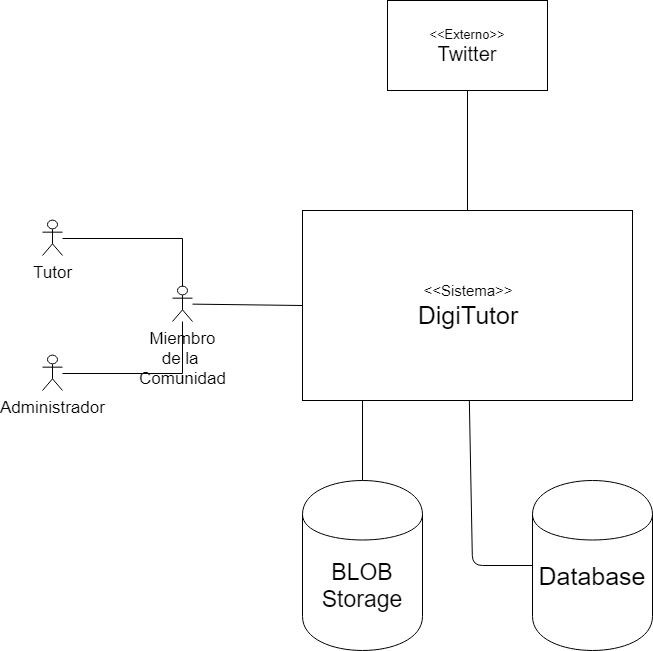
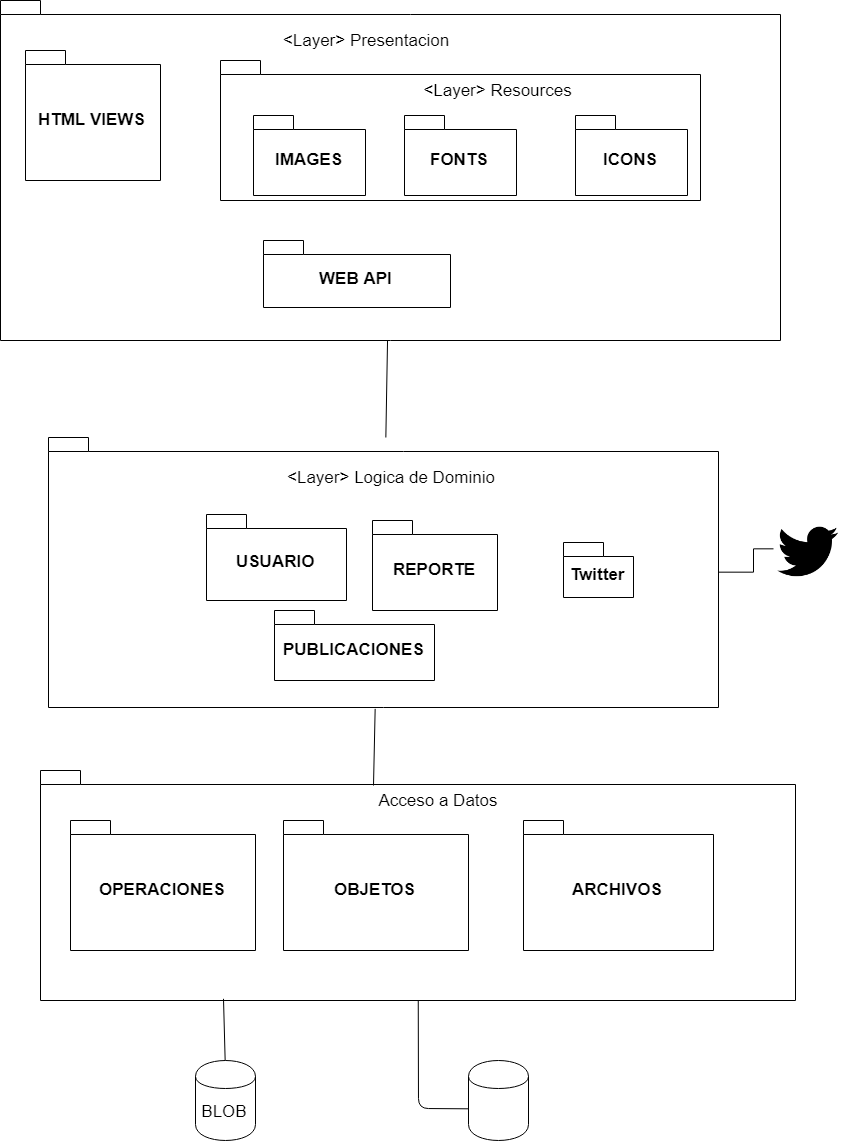


Tabla de descripción del Diagrama de Contexto

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Digi-Tutor | Tutor | Administrador | DataBase | Twitter | Blop Storage |
| Digi-Tutor | N/A | N/A | N/A | El sistema utiliza la base de datos para almacenar los datos de los usuarios. | Digitutor publica en twitter cuando un tutor le da un apoyo a otro tutor en alguna de sus habilidades. | Utiliza el sistema de almacenamiento de Azure para guardar los documentos de las publicaciones de los tutores |
| Tutor | Puede comparir contenido academico y darse a conocer. | N/A | N/A | N/A | N/A | N/A |
| Administrador | Puede agregar categorias, países y universidades ademas de generar reportes. | N/A | N/A | N/A | N/A | N/A |
| DataBase | N/A | N/A | N/A | N/A | N/A | N/A |
| Twitter | N/A | Informa a los miembros de la comunidad de que un tutor dió un apoyo a otro tutor en una de sus habilidades. | N/A | N/A | N/A | N/A |
| Blop Storage | N/A | N/A | N/A | N/A | N/A | N/A |

### Modelo de estructura modular



### **Descripción de capas del sistema Digi-Tutor**

**Capa de presentación:** Permite la interacción del usuario con el sistema Digi-Tutor, se encarga de mostrar información al miembro de la comunidad por medio de distintas vistas y además obtiene los datos de entrada proporcionados por el usuario. Está compuesta por un proyecto en HTML5 y AngularJS, además de un Web Api desarrollado en C#. Esta capa utiliza el modelo Vista-Controlador.

**Capa de lógica de dominio:** En esta se encuentra la lógica necesaria para cumplir con los servicios brindados en la capa de servicios, para esto esta utiliza los datos proporcionados por el usuario a través de la capa de servicios y los datos proporcionados por la capa de acceso de datos. Esta capa está realizada en C#.

**Capa de acceso de datos:** Permite establecer conexión con la base de datos del sistema Digi-Tutor, además en esta capa se establece la conexión con el sistema Twitter necesaria para realizar publicaciones en la cuenta oficial de Digi-Tutor en Twitter. Además esta le permite al sistema Digi-Tutor establecer conexión con el sistema de almacenamiento Blob.

## Requerimientos Funcionales

### Diagrama de Casos de Uso.

VER ANEXOS.

### Diagramas de Secuencia de clases.

VER ANEXOS.

### Diagrama de Clases de Diseño.

### Descripción de patrones de diseño utilizados.

**Patrón Alta Cohesión**

Este se emplea el Web Api, las clases elaboradas en este proyecto de C# presentan alta cohesión, pues las funciones que se realizan en cada clase tienen un mismo enfoque, pues realizan funciones relacionadas.

**Patrón Adapter**

Se tiene un proveedor en el Web API que interpreta un httpcontent permitiendo obtener la información de un httprequest y almacenarlo en el sistema de almacenamiento de archivos de azure

**Patrón Controlador**

En el diseño se crearon clases que sirven como intermediarias entre la interfaz y la lógica que la implementa, estas clases son los controladores, que son las que reciben los datos del usuario y delegan a qué clases deben llamar según el método llamado. Algunas de las clases controladoras presentes en el diseño son las siguiente: Admin Controller, Publicacion Controller, Comments controller, Feed Controller, Universidad Controller, ETC.

## Diseño de Datos.

### Diagrama de Base de Datos

## ANEXOS

