**TRADUCTOR ESPAÑOL-QUECHUA**

**Huaraca, Flores José**

**Llacchuas Alhuay M. Leonel**

***Universidad Nacional José María Arguedas***

***“un hombre sin sueños es un hombre sin futuro.”***

**Abstract**

**En este trabajo se investiga y busca la implementación de un software aplicativo que funcione como un traductor español-quechua (idioma originario nativo), este un software es el principio para hacer nuevas aplicaciones para en ámbito de tecnologías aplicados a la comunicación entre comunidades de poblaciones que hablan diferentes idiomas nativos dentro de nuestro país (Perú).**

**El objetivo de la implementación del software como un aplicativo es que facilite a las personas a aprender a cierto nivel palabras en quechua que no saben para poder comunicarse mejor con otras personas que manejan ese idioma, que así funcione como un medio de aprendizaje y herramienta de ayuda por medio de consulta rápida.**

**El trabajo de investigación busca revalorar el idioma nativo quechua para que las personas busquen aprender más y no se pierda por falta de difusión (que es el problema principal que debemos atacar).**

**El software de aplicación se realizó con el modelo mvc(modelo-vista-controlador) que está orientado a objetos, y el gestor de base de datos utilizado fue el SQLSERVER, desarrollado el visual estido.net.**

**Palabras Claves**

**Software, aplicativo, modelo mvc, español-quechua, visual studio.net, sqlServer, Perú, traductor.**

**Abstract**

**This paper investigates and seeks the implementation of a software application that works as a translator Spanish-Quechua (native language native), this is the first software to new applications in areas of technologies applied to the communication between communities of populations who speak different native languages ​​in our country (Peru).**

**The objective of the implementation of such an application is software that facilitates people to learn at some level in Quechua words they do not know in order to better communicate with others using that language, so functions as a learning tool and aid through quick reference.**

**The research seeks to revalue the native Quechua language for people seeking to learn more and not perish for lack of diffusion (which is the main problem we must attack).**

**The application software was performed with the MVC model (Model-View-Controller) which is object-oriented and database manager was used SQLSERVER, developed the visual estido.net.**

**Keywords**

**Software, application, mvc pattern, Spanish-Quechua, visual studio.net, SQLServer, Peru, translator.**

1. **Introducción**

Por el medio de este proyecto se desea analizar, estudiar y comprender la problemática de la necesidad de comunicarnos ente las personas en diferentes idiomas como por ejemplo la comunicación entre las personas quechua hablantes y las personas que hablan el español como si fuera una cosa cotidiana, pero en la realidad de nuestro país y nuestra región básicamente no pasa eso porque se está perdiendo el lenguaje nativo quechua porque cada generación de personas hablan menos el quechua por la falta de difusión de la lectura y escritura por parte de las instituciones pertinentes que deben ser las primeras en difundir el idioma nativo quechua.

La razón es que en las escuelas no se les enseña a los niños en su idioma de nacimiento si no en español eso hace que los niños olviden a hablar el quechua y más hablen el castellano, su difusión también ayudara a que no haya más discriminación entre las personas de la parte urbana con las personas de la parte de las zonas rurales porque al aprender el quechua también aprendemos a revalorar nuestras costumbres de interculturalidad.

En los últimos tiempos, la tecnología de información se ha convertido es un instrumento de uso diario para muchas personas, el cual es comparable con otros medios tales como la radio, la televisión y el teléfono, nos facilita la comunicación como el intercambio de información, podemos hacer unas series de actividades utilizando estas tecnologías de información. Todo esto conlleva a un notorio crecimiento en su volumen de datos y a una necesidad de acceder a lo mismo de una manera rápida y precisa.

El traductor cumple un rol importante en facilitar el acceso de la información que contiene la base de datos, después de un estudio realizado, se ha dedicado a resolver las necesidades básicas del usuario como el problema al tratar de traducir una palabra que está en el idioma castellano al quechua, pero esta herramienta tecnológica esta para facilitar esta tarea por que el usuario al tener el programa en su computador va a tener la herramienta de traducción a la mano con lo que se le facilitara el aprendizaje de alguna manera. Esta tecnología nos permitirá la utilización de la tecnología como medio de consulta y aprendizaje un momento apropiado y preciso.

Las tecno logias utilizadas para desarrollar el software como un aplicativo (traductor español-quechua), son herramientas tecnológicas que están a nuestro alcance por no son tan costosas su adquision estas herramientas son el gestor de base de datos sqlServer, visual estudio.net en donde se desarrolla de fácil la programación de aplicaciones.

**1.2 problema central**

El problema central que pretendemos abordar es:

Perdida de la difusión del idioma quechua como una fuente de comunicación entre las personas de nuestra región.

**1.3 objetivos**

* **Implementación de un software de ayuda para la traducción de palabras en castellano al quechua**
* **Utilización de herramientas nuevas para la utilización en la programación**

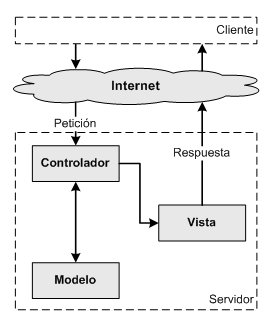
1. **Métodos**

**2.1 herramientas y técnicas utilizadas**

**Patrón Modelo Vista Controlador**

El patrón Modelo – Vista – Controlador fue inventado en el contexto de Smalltak para realizar una separación entre la interfaz gráfica y el código del funcionamiento de una

Aplicación. Esta idea teórica afectó, de forma importante, a gran parte del código de SmallTak y fue posteriormente aplicada a los lenguajes que están basados en objetos.



**Figura1.representacion gráfica del modelo–vista-controlador**

**Modelo**

El modelo es un conjunto de clases que representan la información del mundo real que el sistema debe reflejar. Es la parte encargada de representar la lógica de negocio de una aplicación. Así, por ejemplo, un sistema de administración de datos geográficos tendrá un modelo que representara la altura, coordenadas de posición, distancia, etc. sin tomar en cuenta ni la forma en la que esa información va a ser mostrada ni los mecanismos que hacen que esos datos estén dentro del modelo, es decir, sin tener relación con ninguna otra entidad dentro de la aplicación.

El modelo, a nivel teórico, no debe tener conocimiento acerca de la existencia de las vistas y del controlador. Esta situación es interesante, pero de difícil aplicación práctica, pues deben existir interfaces que permitan a los módulos comunicarse entre sí, por lo que SmallTalk sugiere que el modelo en realidad esté formado por dos submódulos: El modelo del dominio y el modelo de la aplicación.

**Modelo de la aplicación**

El modelo de la aplicación es un conjunto de clases que sirven de puente en la relación de las vistas con el modelo de dominio. Tienen conocimiento de las vistas e implementan los mecanismos necesarios para notificar a éstas los cambios que se pudieren dar en el modelo del dominio. El modelo de la aplicación es llamado también coordinador de la aplicación.

**Vista**

Las vistas son las encargadas de la representación de los datos, contenidos en el modelo, al usuario. La relación entre las vistas y el modelo son de muchas a uno, es decir cada vista se asocia a un modelo, pero pueden existir muchas vistas asociadas al mismo modelo. De esta manera, por ejemplo, se puede tener una vista mostrando la hora del sistema como un reloj analógico y otra vista mostrando la misma información como un reloj digital.

La vista solo necesita la información requerida del modelo para realizar un despliegue. Cada vez que se realiza una actuación, que implica una modificación del modelo de dominio, la vista cambia a través de notificaciones generadas por el modelo de la aplicación. Sencillamente, es la representación visual del modelo que redibuja las partes necesarias cuando se produce una modificación del mismo.

**Controlador**

El controlador es el encargado de interpretar y dar sentido a las instrucciones que realiza el usuario, realizando actuaciones sobre el modelo. Si se realiza algún cambio, comienza a actuar, tanto si la modificación se produce en una vista o en el modelo. Interactúa con el Modelo a través de una referencia al propio Modelo.

**¿Cómo funciona una aplicación MVC?**

**Captura de la petición en el controlador**

La aplicación recibe peticiones que son centralizadas en el Controlador. Éste es el encargado de interpretar, a partir de la URL de la solicitud, el tipo de operación que hay que realizar. Normalmente, esto se hace analizando el valor de algún parámetro que se envía anexando a la URL de la petición y que se utiliza con esta finalidad.

**Procesamiento de la petición**

Una vez que el Controlador determine la operación a realizar, procede a ejecutar las acciones pertinentes, invocando para ello a los diferentes métodos expuestos por el Modelo.

Dependiendo de las acciones a realizar (por ejemplo, un alta de un usuario en el sistema), el Modelo necesitará manejar los datos enviados por el cliente en la petición, datos que le serán proporcionados por el controlador. De la misma manera, los resultados generados por el Modelo (por ejemplo la información resultante de una búsqueda serán entregados directamente al controlador).

**Generación de respuestas**

Los resultados devueltos por el Modelo al Controlador son depositados por éste en una variable de petición, sesión o aplicación, según el alcance que deban tener. A continuación, el Controlador invoca a la página JSP que debe encargarse de generar la vista correspondiente, está página accederá a la variable de ámbito donde estén depositados los resultados y los utilizará para generar dinámicamente la respuesta XHTML que será enviada al cliente.

**Microsoft SQL Server**

Es un sistema para la gestión de bases de datos producido por Microsoft basado en el modelo relacional. El Motor de base de datos es el servicio principal para almacenar, procesar y proteger los datos. El Motor de base de datos proporciona acceso controlado y procesamiento de transacciones rápido para cumplir con los requisitos de las aplicaciones consumidoras de datos más exigentes de su empresa.

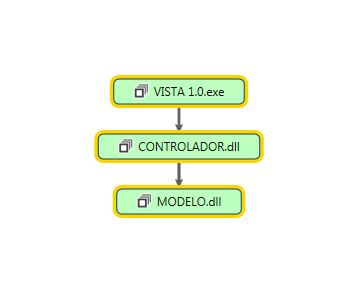
Use Motor de base de datos para crear bases de datos relacionales para el procesamiento de transacciones en línea o datos de procesamiento analíticos en línea. Se pueden crear tablas para almacenar datos y objetos de base de datos como índices, vistas y procedimientos almacenados para ver, administrar y proteger los datos. Puede usar SQL Server Management Studio para administrar los objetos de bases de datos.

**Visual Basic .NET**

es un lenguaje de programación orientado a objetos que se puede considerar una evolución de Visual Basic implementada sobre el framework .NET. Su introducción resultó muy controvertida, ya que debido a cambios significativos en el lenguaje VB.NET no es retro compatible con Visual Basic, pero el manejo de las instrucciones es similar a versiones anteriores de Visual Basic, facilitando así el desarrollo de aplicaciones más avanzadas con herramientas modernas. Para mantener eficacia en el desarrollo de las aplicaciones. La gran mayoría de programadores de VB.NET utilizan el entorno de desarrollo integrado Microsoft Visual Studio en alguna de sus versiones (desde el primer Visual Studio .NET hasta Visual Studio .NET 2013, que es la última versión de Visual Studio para la plataforma .NET).

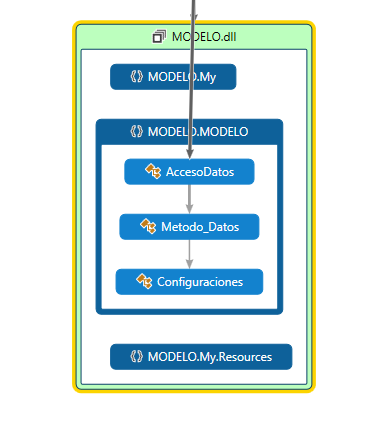
**DESCRIPCION DEL DESARROLLO DEL SOTFWAR (traductor español-quechua orientado a la escritura delos números)**

Esta la descripción del desarrollo del software de aplicación que se basa en la utilización del modelo de programación por capas orientado a objetos Modelo-Vista-Controlador que se utilizó en el desarrollo

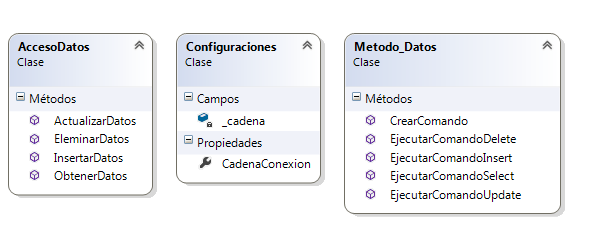


**Figura2.vista grafica denlas capas de los modelos utilizados**

En la capa de modelo se tiene las conexiones a diferentes componentes del programación como son las clases de acceso a datos, configuraciones de conexión, método-de clases su principal función es la conexión al gestor de base de datos y la manipulación de los datos dentro del gestor de base de datos.

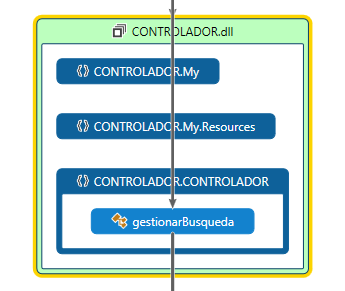


**Figura3.descripcion grafica de la capa de modelo**

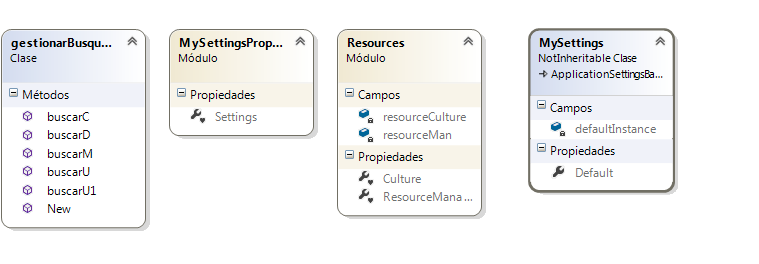


**Figura4. Vista grafica de las clases de la capa de modelo**

En la capa del controlador tenemos los componentes de búsqueda, así como también acceso a los recursos (donde podemos controlar los procesos que desarrollemos dentro del gestor de búsqueda).

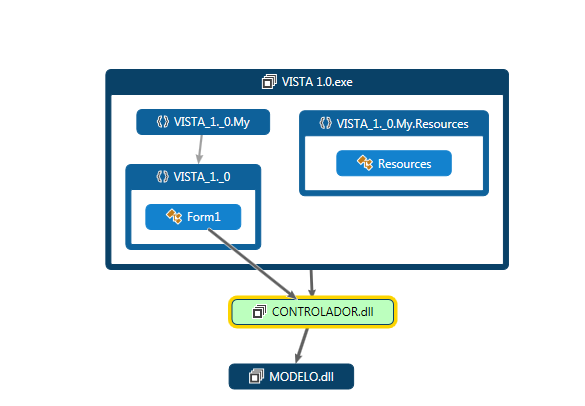


**Figura5.vista grafica de la capa del controlador**

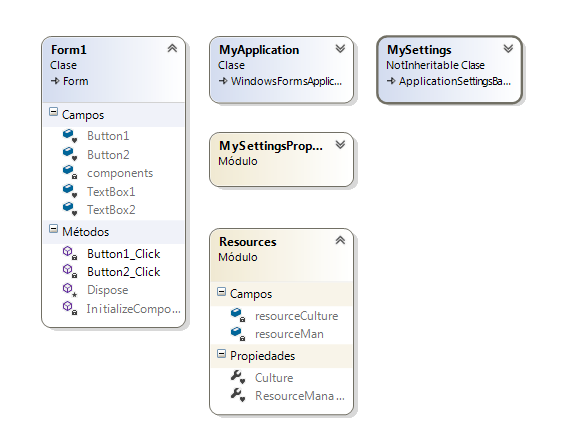


**Figura6.vista grafica delos componentes de la capa del controlador**

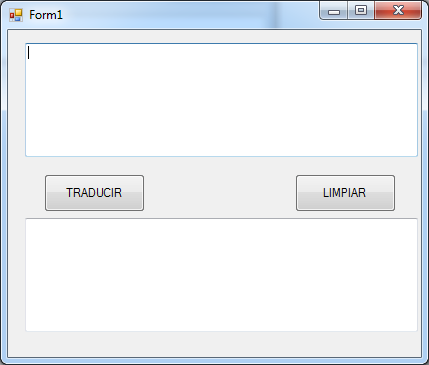
En la capa de la vista tenemos el formulario de interfaz en donde se implementara todos los botones (métodos ),y la interfaz más adecuada donde el usuario final interactuara con el software que implementamos



**Figura7.vista grafica de la capa de la vista**



**Figura8.vista grafica de los componentes de la capa de la vista**



**Figura9.vista del interfaz que vera el usuario final para sus consultas**

Estos son los modelos no relacionales que necesitamos implementar para crear la base de datos que guarde los datos con los que trabajaremos y que debemos crear en el sqlServer (el gestor de base de datos con el cual trabajamos la aplicación)

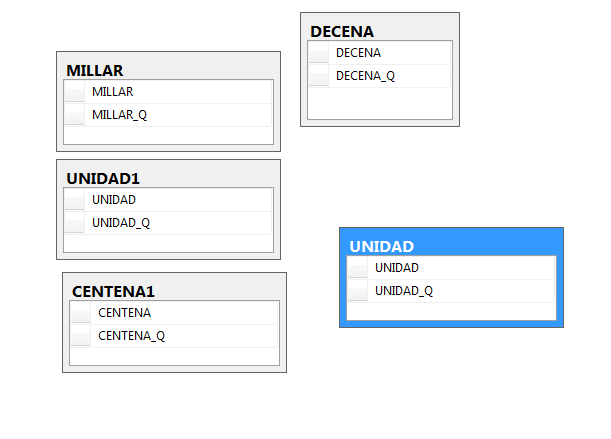


Figura10.vista grafica de modelo no relacional

1. **Resultados**

* Los usuarios finales no tendrán ningún problema en la utilización de software porque está hecho de manera que el usuario manipule de forma fácil.
* Las pruebas que se hicieron en el momento de la implementación del software sirvieron para arreglar algunas deficiencias que se presentaron al momento de hacer algunas implementaciones adiciónales en la aplicación
* El modelo de programación MVC (modelo-vista-controlador ) nos ayudó a hacer la implementación del software más entendible donde se encuentran los errores en la programación.

1. **Conclusión**

El objetivo de este trabajo era implementar un traductor español quechua que pueda ser reutilizable para seguir trabajando, de manera que su código fuente pueda ser reutilizable.

El leguaje nativo quechua tiene una sintaxis muy diferente a la del español y hace la traducción muy difícil, pero hemos hecho nuestro mejor esfuerzo en desarrollar la traducción de las palabras más comunes como así también de los números que se encuentran en la base de datos utilizado , tanto su numeración en español, y su traducción en quechua.

Así concluimos en que nos gustaría seguir desarrollando más adelanté este trabajo de una forma más amplia porque nos ha dejado con bastantes preguntas por investigar, por qué es un tema más amplio que se debe de abordar como un software de aplicación a la educación (como un software de autoaprendizaje, y herramienta tecnológica que nos va ayudar a conocer más la lengua nativa quechua).

1. **Agradecimiento**

El presente trabajo simboliza una etapa importante de mi vida y materializa todo el esfuerzo realizado para alcanzar la profesión por la que tanta vocación siento. Todo esto no hubiera sido posible sin la ayuda de las personas que me rodean y a las cuales no quiero dejar de agradecer. A mi familia por inculcarme la importancia de tener una profesión y generar el desafío personal por alcanzarla. A los profesores que me formaron durante todos estos años de estudio, mis compañeros de estudio, por las revisiones, aportes de ideas y horas de estudio compartidas. Al Ing. Iván Soria por el esfuerzo dedicado como tutor de la presente proyecto.

1. **Referencia**

[1] <http://www.dc.fi.udc.es/muc/sites/www.dc.fi.udc.es.muc/files/ValUsab.pdf>

[2]<http://msdn.microsoft.com/es-es/magazine/cc337884.aspx>

[3]<http://social.msdn.microsoft.com/Forums/es-ES/aspnetmvces/threads>

**[4]**[**http://www.programacionfacil.com/visual\_basic\_net/start**](http://www.programacionfacil.com/visual_basic_net/start)

**[5]**[**http://www.canalvisualbasic.net/manual-net/vb-net/**](http://www.canalvisualbasic.net/manual-net/vb-net/)

**[6]**[**http://www.mundoprogramacion.com/default.aspx**](http://www.mundoprogramacion.com/default.aspx)

**[7]**[**http://msdn.microsoft.com/es-es/library/hh231699.aspx**](http://msdn.microsoft.com/es-es/library/hh231699.aspx)

**[8]**[**http://dono.discapnet.es/node/1165**](http://dono.discapnet.es/node/1165)

**[9]**

[**http://www.sitcorp.com/wp-content/uploads/2012/10/SQL\_Server\_2012\_Licensing\_Guide.pdf**](http://www.sitcorp.com/wp-content/uploads/2012/10/SQL_Server_2012_Licensing_Guide.pdf)

1. **Datos de Contacto**

***José huaraca flores***

[***josshf16@gmail.com***](mailto:josshf16@gmail.com)

***Numero de celular 958107541***

**M Leonel Llacchuas Alhuay**

[**Alhuay-17@hotmail.com**](mailto:Alhuay-17@hotmail.com)

**Cel. 983900489**