

**FACULTAD DE INGENIERÍA**

**ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA DE SISTEMAS**

**CALIDAD DE SOFTWARE**

CONTROL DE CALIDAD DEL SOFTWARE DE LA I.E. ANTONIO RAYMONDI

**Integrantes:**

Espinoza Hernandez, Alexandra

Romero Sandoval, Elver

Alvarez Leyva, Cristopher

**Asesor:**

Jorge Alfredo Guevara Jimenez

**Lima – Perú**

**2017**

ÍNDICE

### INTRODUCCIÓN………………………………………………………………………………………..3

### MARCO TEÓRICO………………………………………………………………………………………5

**DESCRIPCIÓN GLOBAL DEL SOFTWARE……………………………………………………..5**

**CONCLUSIONES………………………………………………………………………………………..13**

**RECOMENDACIONES………………………………………………………………………………..14**

**REFERENCIAS BIBLIOGRÁFICAS………………………………………………………………………………………15**

### INTRODUCCIÓN

En el presente informe se llevará a cabo como el software que está ligado en el entorno web de la institución Antonio Raymondi cumple con las características de calidad, así como su funcionalidad, compatibilidad, fiabilidad y entre otras.

Esta inspección tendrá como apoyo los conceptos teóricos y básicos sobre calidad de software; puesto a ello se recurrirá a la documentación del software para la veracidad de los requisitos planteados, prototipos, etc del sistema a tratar.

Finalmente, el software pasara el control de calidad si verdaderamente cumple con las normas y estándares de calidad.

### MARCO TEÓRICO

* **Calidad de Software**

Según Vega *e.t* (2008) definen a calidad de software “[..]La totalidad de características de un producto, proceso o servicio que cuenta con la habilidad de satisfacer necesidades explícitas o implícitas”.

Por otro lado, Fernandez *e.t* (2008) detalla que “la calidad del software es el conjunto de cualidades que lo caracterizan y que determinan su utilidad y existencia. La calidad es sinónimo de eficiencia, flexibilidad, corrección, confiabilidad, mantenibilidad, portabilidad, usabilidad, seguridad e integridad”.

* **Calidad de producto y proceso**

Según Vallejo, Cathalina (2013) define que la calidad de producto y proceso es “El aseguramiento de calidad de proceso y productos software, ofrece un conjunto de prácticas fundamentadas en normas de calidad internacionales, cuyo foco radica en sembrar en los Proyectos, la confianza de que sus productos satisfacen los requisitos de calidad que esperan percibir los usuarios.

* **Calidad Interna**

Según Vialfa, Carlos (2017) define “la calidad interna como al mejoramiento de la operación interna de una compañía. El propósito de la calidad interna es implementar los medios para permitir la mejor descripción posible de la organización y detectar y limitar los funcionamientos incorrectos. Los beneficiarios de la calidad interna son la administración y los empleados de la compañía”.

* **Calidad Externa**

Según Vialfa, Carlos (2017) detalla que “corresponde a la satisfacción de los clientes. El logro de la calidad externa requiere proporcionar productos o servicios que satisfagan las expectativas del cliente para fidelizarlos y de ese modo mejorar su participación en el mercado. Los beneficiarios de la calidad externa son los clientes y los socios externos de una compañía”.

* **Calidad en uso**

Medina, Gustavo (2014) define la calidad en uso como “[…]la percepción que tiene el usuario en la experiencia de uso del software, el cual depende de manera sustancial tanto del producto como del entorno en el cual se ejecuta.”

* **Norma ISO 25010**

Según la ISO 25000 (2015) define la norma ISO 25010 como “la calidad del producto software se puede interpretar como el grado en que dicho producto satisface los requisitos de sus usuarios aportando de esta manera un valor. Son precisamente estos requisitos (funcionalidad, rendimiento, seguridad, mantenibilidad, etc.) los que se encuentran representados en el modelo de calidad, el cual categoriza la calidad del producto en características y subcaracterísticas”.

* **Norma ISO 12207**

Según Cornejo, Alejandro (2015) define a la ISO 12207 como la encargada “[…] de todo lo relacionado al ciclo de vida del software, desde la conceptualización de ideas hasta la retirada y consta de procesos para la adquisición y suministro de proyectos y servicios del software, estableciendo pautas para su control y mantenimiento”. También, especificando que como su objetivo de la Norma ISO 12207 es “[…] proporcionar una estructura común para que los compradores, proveedores, desarrolladores, personal de mantenimiento, operadores, gestores y técnicos involucrados en el desarrollo de software utilicen un lenguaje en común”.

* **Procesos del Software (Ciclo de Vida)**

Según Ruvalcaba, Mara (2004) expresa que el proceso del software es “[…] un conjunto de personas, estructuras de organización, reglas, políticas, actividades y sus procedimientos, componentes de software, metodologías, y herramientas utilizadas o creadas específicamente para definir, desarrollar, ofrecer un servicio, innovar y extender un producto de software.

* **Guía PMBOK**

Según EAE Bussiness School (2014) expresa que “La guía del PMBOK es un instrumento desarrollado por el Project Management Institute (o PMI), que establece un criterio de buenas prácticas relacionadas con la gestión, la administración y la dirección de proyectos mediante la implementación de técnicas y herramientas que permiten identificar un conjunto de 47 procesos, distribuidos a su turno en 5 macroprocesos generales”.

* **Proceso unificado**

Según Jesus, Melania (2010) explica que “El proceso unificado (UP, o Unified Development Process) es una versión libre y abierta (no propietaria) del proceso iterativo e incremental de ingeniería de software propuesto por Jacobson, Booch y Rumbaugh (los “tres amigos”) en su libro El proceso unificado de desarrollo de software, publicado por Addisson-Wesley en 1999. El lenguaje para especificar y diagramar en el UP es UML, por lo cual puede apoyarse en cualquier herramienta que soporte UML. Sus características principales son: Está dirigido por casos de uso (véase la sección sobre UML).Está centrado en la arquitectura (es decir, en una solución de conjunto. Tiene un ciclo de vida iterativo incremental”.

### 

### DESCRIPCIÓN GLOBAL DEL SOFTWARE

**Objetivos:**

|  |  |
| --- | --- |
| **ID** | **Descripción** |
| ON-1 | Reducir en un 40% la espera de los padres o apoderados para la matrícula. |
| ON-2 | Reducir en un 40% la espera en pagos en tesorería |
| ON-3 | Reducir en un 50% el mal manejo de los documentos de los alumnos y guardarlos en la nube. |

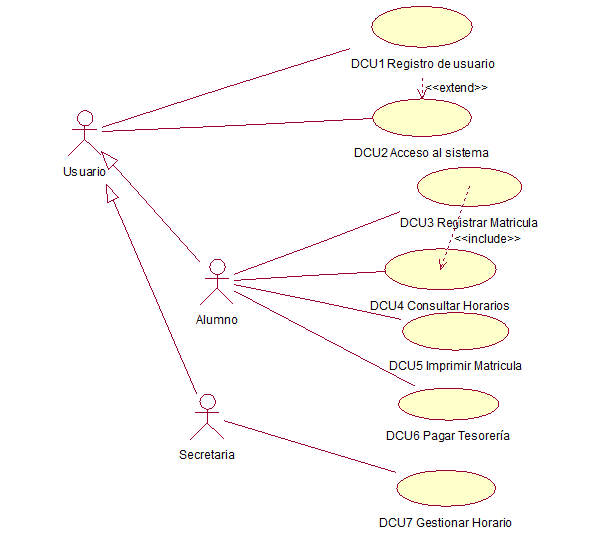
**DATOS DE LA EMPRESA:**

Institución beneficiaria**:** COLEGIO ANTONIO RAYMONDI

RUC: 20601952182

Dirección: JR. TUPAC AMARU NRO. 475

Contacto: Elver Romero

**Diagramas de casos de uso:**

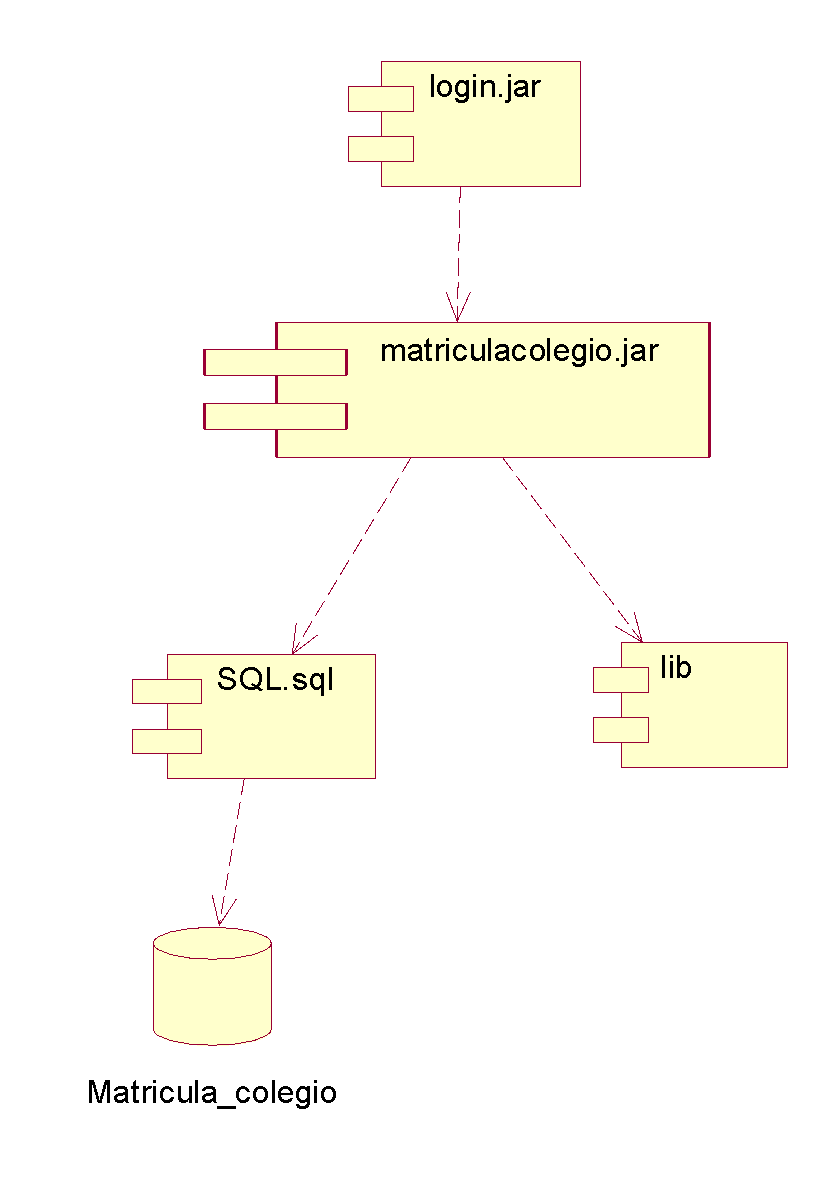
### Requisitos Funcionales:

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Descripción | Prioridad | Objetivo  Negocio |
| RF-1 | El sistema debe matricular a los alumnos mediante una aplicación web. | Alta | ON-1 Y ON-2 |
| RF-2 | El sistema debe permitir consultar horarios matriculados. | Alta | ON-1 Y ON-2 |
| RF-3 | El sistema debe imprimir una matrícula. | Alta | ON-1 Y ON-2 |
| RF-4 | El sistema debe registrar el pago de tesorería | Alta | ON-1, ON-2 Y ON-3 |
| RF-5 | El sistema debe permitir gestionar los horarios | Media | ON-1 Y ON-2 |

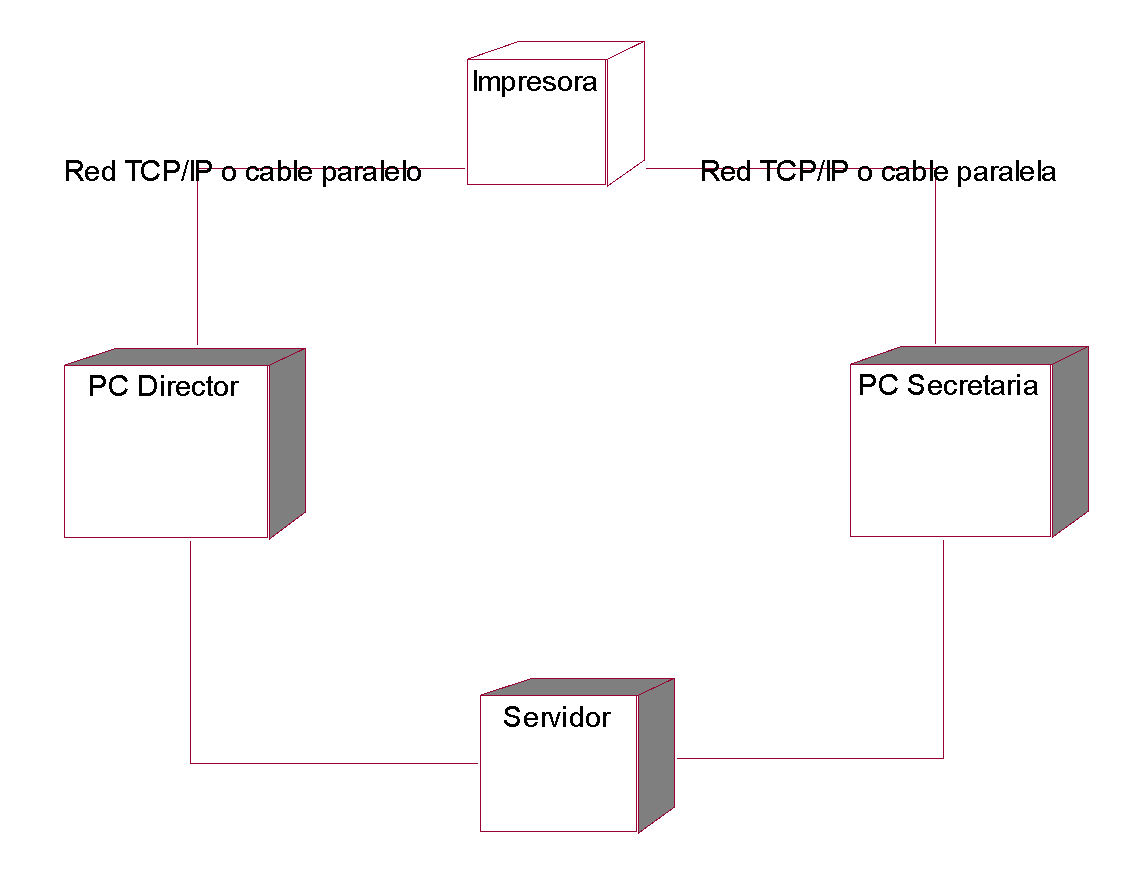
### Requisitos no funcionales:

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Descripción | Prioridad | Objetivo  Negocio |
| RNF-1 | El sistema debe permitir realizar cualquier pago (tesorería u otra deuda pendiente) con tarjeta BCP, BBVA, etc. | Alta | ON-1, ON-2 Y ON-3 |
| RNF-2 | El sistema debe exportar un servicio para sistemas móviles (celular, Tablet, etc.) | Alta | ON-1 Y ON-2 |

### Arquitectura de software:



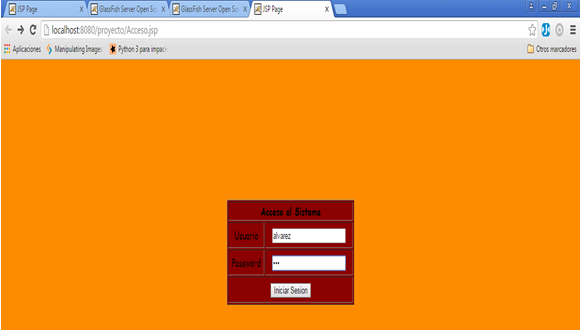
### Arquitectura del Hardware:



### 

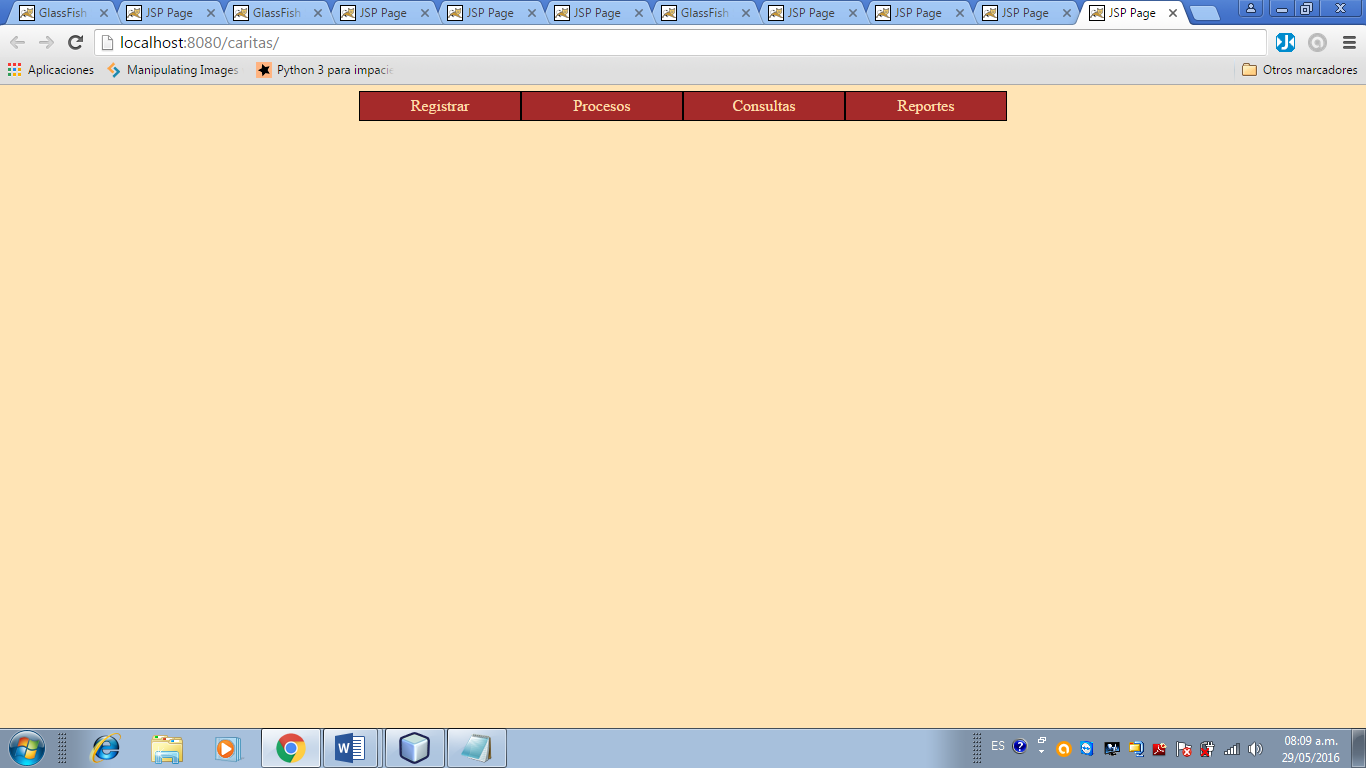
### Diseño de Interfaz Gráfica:

* **LOGIN**



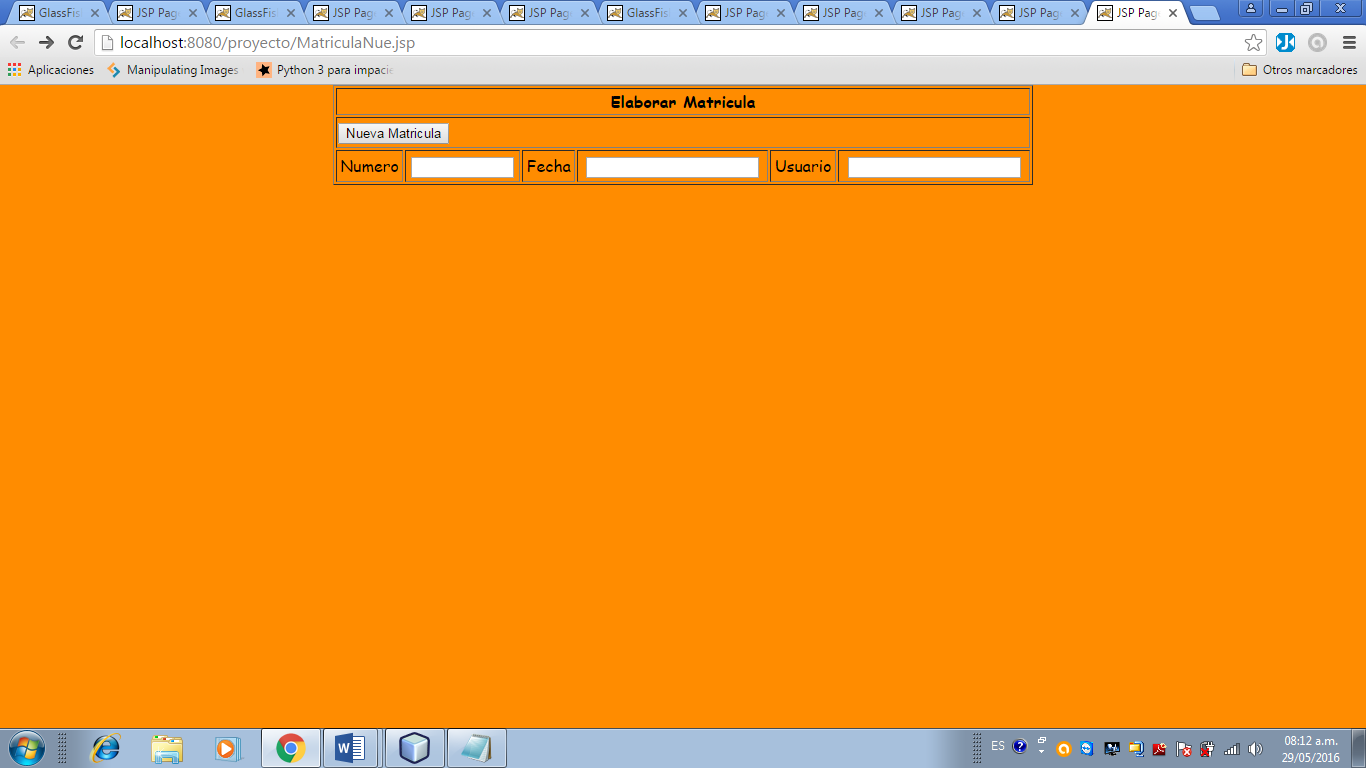
* **Menu Base**

Consta de 4 secciones con sus respectivas alternativas:



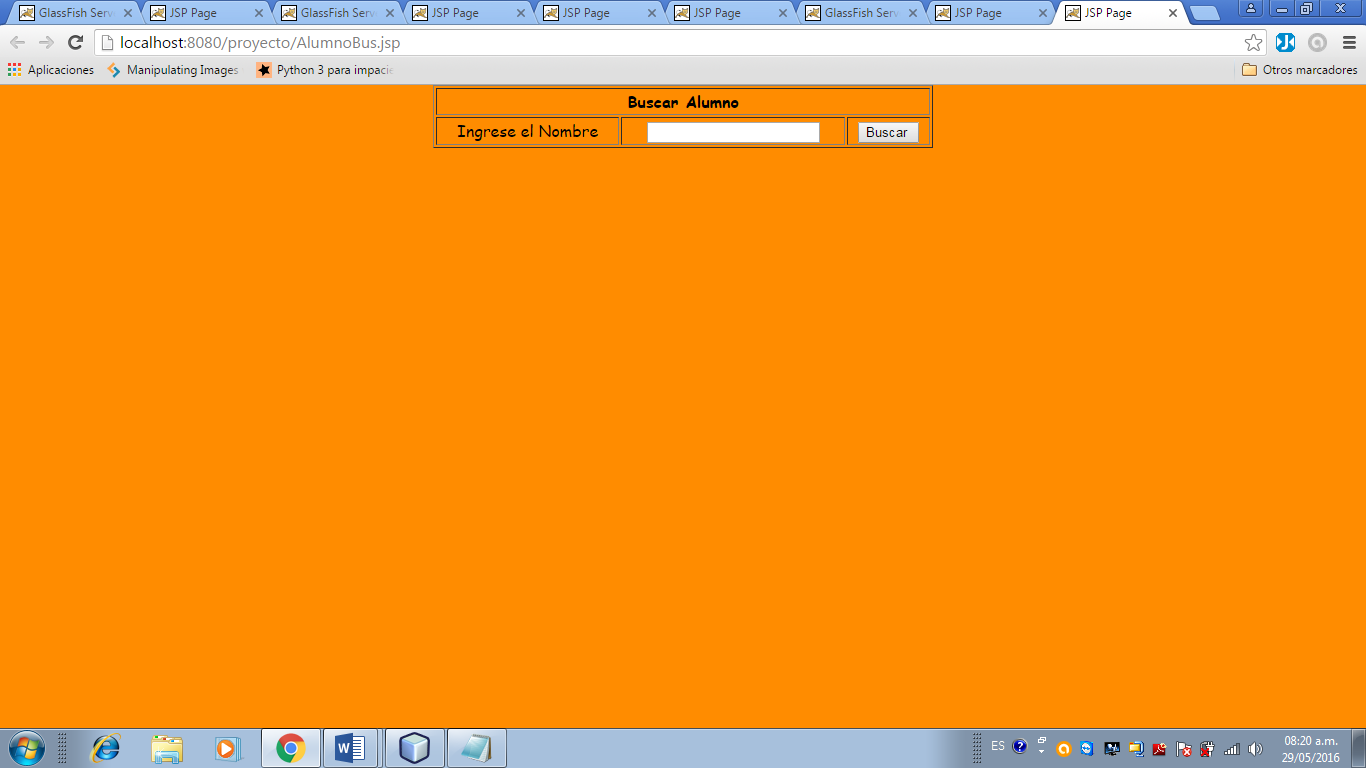
* **Matricula Nueva**

Acceso a una nueva matrícula para seleccionar el alumno el grado y de acuerdo con eso introducir los cursos el profesor y el horario y realizar la matricula.



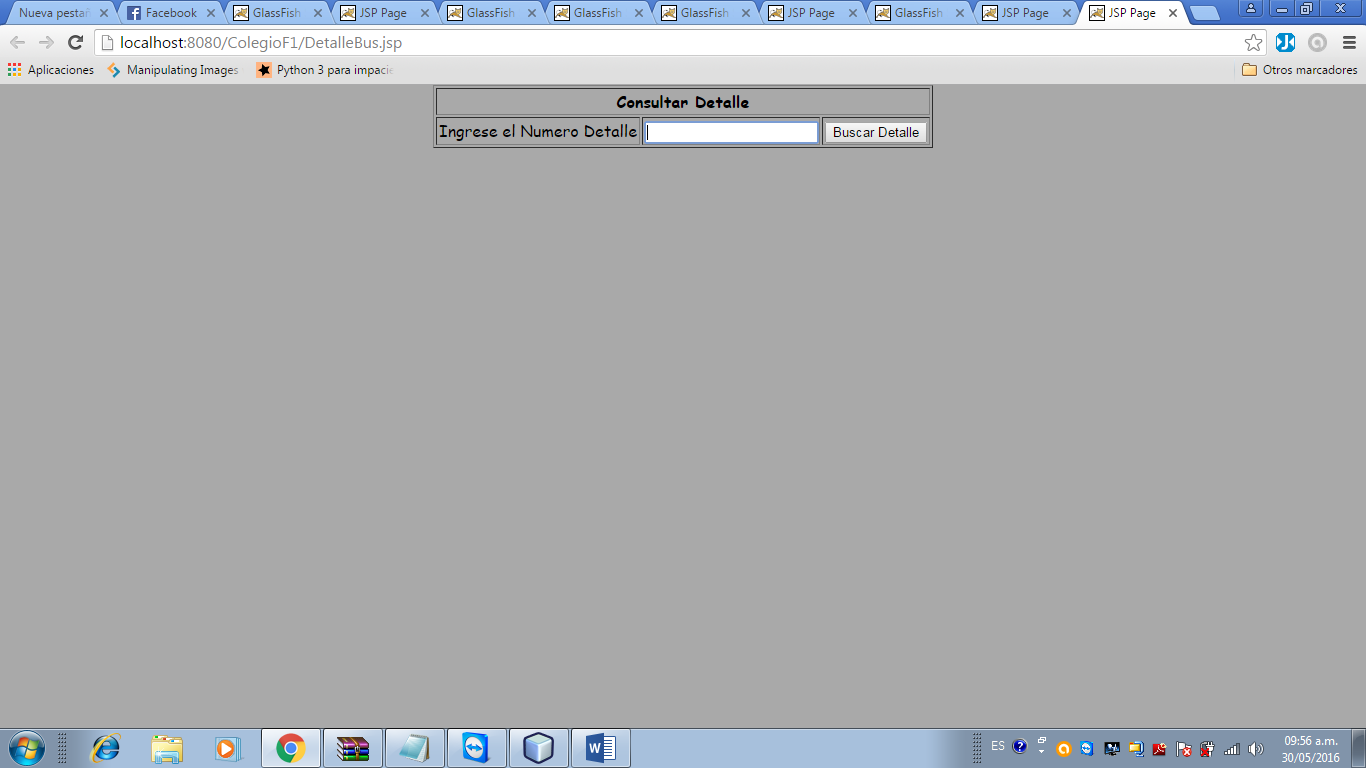
* **Buscar Alumno**

Se introducirá el nombre del estudiante para poder acceder a sus datos e introducirla en la matricula



* **Buscar Detalle**

Se buscará el detalle según para seleccionar todos los datos pertenecientes al grado que pasara el estudiante como cursos profesores aula y horario



### CONCLUSIONES

* En conclusión, este trabajo ayuda a que el producto y la calidad de un software sea de la manera más óptima y cumpla con las necesidades en general de la empresa, está en conjunto con la satisfacción también del cliente.
* Los procesos del software deben soportar o facilitar el cumplimiento de características de calidad que satisfacen los estándares requeridos para los productos software desarrollados.
* Potenciar los procesos que determinan el cumplimiento de las características de calidad del producto que se desarrolla incrementaría la satisfacción del cliente y a su vez la calidad del producto.
* Este trabajo está planteado con medidas realizadas para la calidad del software, como se relaciona el sistema con las características y subcaracterísticas del estándar de calidad.
* El planteamiento del trabajo de calidad de software hace un uso preferible de los estándares para así tener un sistema de calidad y que tenga una interacción con el usuario, como también sea de su beneficio y agrado.

**Conclusiones a la calidad de proceso:**

-Para poder realizar un buen producto es necesario que la calidad del proceso este elaborada de manera correcta y eficaz de tal manera que mediante la documentación se compruebe.

-Hacer una clara definición del sistema de gestión de calidad que se vaya a implantar y establecer, valorando la posibilidad de contar o no con apoyo externo.

-Elaborar un presupuesto y mantener el compromiso de cumplimiento, junto a las necesidades de formación respecto al proyecto de cara a la implantación del sistema.

-Realizar una documentación la cual especifique cada proceso como también cada cambio ocurrido en la elaboración del sistema junto con el funcionamiento de este para el cumplimiento de los objetivos del proyecto.

-Mejorando la calidad de lo que se está haciendo conduce a tener menos fallas, menos coste, productividad más alta, mejor calidad y más satisfacción por parte de todos los involucrados del proyecto.

**Conclusiones a la calidad del código fuente:**

-Realizar medidas las cuales nos den a conocer mejor las características y funcionamiento del software como el tiempo de carga en cada una de sus gestiones.

-Comprobar el buen desarrollo de elaboración del código en cada una de las gestiones hechas según los objetivos propuestos en el proyecto.

-Tras una extensa evaluación al código del sistema, probando mejores formas de hacer de él un fácil uso para el cliente se lograra un eficaz desarrollo de este.

-La calidad de proceso favorece a tener también una mejor evaluación del código elaborado para el sistema en implementación, antes de su ejecución.

-Consultar con la organización para verificar que el sistema esté cumpliendo con sus necesidades, teniendo una buena comunicación.

**Conclusiones a la evaluación de métricas:**

-Las métricas de procesos, productos y recursos son necesarias para tener un preciso conocimiento del software.

-Las métricas ayudan a servir de medio para entender, monitorizar, controlar, predecir y probar el desarrollo del software y los proyectos de mantenimiento de este para su buen desempeño.

-Ayudan a controlar qué es lo que ocurre durante la creación y uso del software en nuestros proyectos.

-Aplicar una medida mejora nuestros procesos y nuestros productos en su desarrollo teniendo así un proyecto el cual cumpla con sus objetivos propuestos.

-Facilitan dar a conocer si se encuentra alguna anormalidad en el sistema durante su funcionamiento, en conjunto con su desempeño para el cliente quien lo controle.

### RECOMENDACIONES

* En general se recomienda que el uso de las características para el desarrollo apropiado del sistema de un software es indispensable y necesario. Ayuda para tener un funcionamiento más exacto cumpliendo las expectativas de la empresa , satisfaciendo también las necesidades del usuario.
* También se recomienda realizar una documentación apropiada teniendo todo ordenado para que de esta manera se puedan identificar los errores fallas y defectos.
* Al identificar los posibles errores se pueden tomar planes de contingencia.
* Se recomienda las pruebas de calidad del software viendo que se realicen por un equipo externo al desarrollo del producto para tener una mejor visión de sus posibles errores.
* Para mejorar la satisfacción del cliente el producto debe contar con una interfaz amigable fácil de entender para el usuario.

**Recomendaciones a la calidad de proceso:**

- Tener en constante mantenimiento mediante diversos tipos de control para así poder conseguir un aumento gradual y constante de la calidad, productividad y eficacia.

- La utilización de la informática como forma de automatización de procesos, detección de errores y propuestas de mejora brindan buenos resultados.

- Se debe tener un buen orden y organización en la aplicación de evaluar la calidad de los procesos para poder tener una apropiada documentación.

-Como parte de la evaluación se debe realizar planificaciones y auditoría de gestión que permitan evaluar los procedimientos de cada parte de los procesos, funciones y unidades, con la finalidad de establecer estrategias y mejoras en la calidad.

-Llevar una buena relación de comunicación junto con la organización, para poder tener una buena implementación como también ejecución al finalizar el proyecto.

**Recomendaciones a la calidad de código fuente:**

-Los procesos de mejora para el software son siempre iterativos y necesitan multitud de ajustes pequeños para ir comprobando cómo afectan estos en los objetivos propuestos en el proyecto como por ejemplo, la disminución de la aparición de un cierto tipo de defectos.

-Ciertos defectos suelen ser detectados en el momento de la ejecución (como las fugas de memoria), así que lo mejor es tener un apropiado plan de contingencia contra esto.

-Tener una herramienta que ayude a dar soporte, entre otras tareas, que ayude al análisis de la calidad del código es muy eficaz ya que ayuda de mejor manera reportando los problemas a los interesados y permitiendo que exista un criterio común de cómo se deben hacer las cosas respecto a los objetivos a alcanzar.

-Es bueno conservar versiones anteriores (backups) del sistema para poder restaurar así este ante cualquier falla en una actualización.

-Notificar cada modificación o cambio realizado en el código, detalladamente en la documentación para poder tener una mejor organización y orden en la elaboración de este software.

**Recomendaciones a la evaluación de métricas:**

-Desarrollar métricas que permitan evaluar el cumplimiento de metas y objetivos del sistema en cada una de sus gestiones ayuda a su buen desempeño para la satisfacción del cliente.

-Es importante cuantificar la pérdida de tiempo que un proceso genera en su labor a través de los resultados que arrojen los indicadores de gestión de la productividad.

-Mostrar el presupuesto del proyecto a realizar es necesaria mediante la evaluación de las métricas hechas de cada parte de planificación, implementación y ejecución de los procesos.

-Analizar el tiempo de desarrollo empleado es necesaria para poder estar previstos ante cualquier variación de este, teniendo un plan de contingencia.

-Evitar desviaciones en el alcance, dando un control y supervisión diario de que el alcance propuesto sea logrado mediante todos los procesos previstos sin ninguna acción fuera de la prevista.

**REFERENCIAS BIBLIOGRÁFICAS**

Cathalina Vallejo (2013). *“Servicio de Calidad de Proceso y Producto Software”.* Recuperado de: <http://www.harriague.com/novedades/servicio-de-calidad-de-proceso-y-producto-software/>

Gustavo Medina (2014). “*Calidad en uso: Definición y evaluación de un modelo de calidad en uso utilizando la norma ISO/IEC 25000 (Spanish Edition)”.* Recuperado de: <https://www.amazon.com/Calidad-uso-Definici%C3%B3n-evaluaci%C3%B3n-utilizando/dp/3659022179>

ISO 25000 (2015). *“ISO/IEC 25010”.* Recuperado de: <http://iso25000.com/index.php/normas-iso-25000/iso-25010>

Vega *e.t* (2008).”Calidad de un software”. Recuperado de:

<http://imgbiblio.vaneduc.edu.ar/fulltext/files/TC110592.pdf>

Fernandez *e.t* (2008).”Organización de una empresa” Recuperado de:

<https://www.yeastgenome.org/reference/S000127432>

Vialfa, Carlos (2017).”Las TIC y el uso indispensable del software”. Recuperado de:

<http://es.ccm.net/forum/affich-1756564-como-descargar-libros>

EAE Bussiness School (2014). “Qué es la guía PMBOK y cómo influye en la administración de proyectos”. Recuperado de: <https://retos-operaciones-logistica.eae.es/que-es-la-guia-pmbok-y-como-influye-en-la-administracion-de-proyectos/>

Ruvalcaba, Mara (2004). “Procesos de software”. Recuperado de: <https://sg.com.mx/revista/1/procesos-software#.We59TWjWzDc>

Jesus, Melania (2010). “Gestión de Sistemas”. Recuperado de: <https://mdjesus.wordpress.com/2010/05/19/84/>

Cornejo, Alejandro (2015). “ISO 12207”. Recuperado de: <https://normasyestandaresproyectosti.wordpress.com/2015/01/29/iso-12207/>