



**UDH**  
UNIVERSIDAD DE HUÁNUCO

**UNIVERSIDAD DE HUÁNUCO**  
Escuela de Post Grado

# **Maestría en Ingeniería de Sistemas e Informática**

## **TESIS**

APLICACIÓN DE PSP (PERSONAL SOFTWARE PROCESS)  
PARA EL DESARROLLO DEL SISTEMA DE BOLSA DE  
TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO - 2016.

Para Optar el Grado Académico de:  
**MAESTRO EN INGENIERÍA DE SISTEMAS E  
INFORMÁTICA**

**Mención en Gerencia de Sistemas y Tecnologías de la Información**

### **AUTOR**

SILVERIO BRAVO, Fernando Fortunato

### **ASESOR**

Dr. BERNARDO TELLO, Alcides

**Huánuco -Perú**  
**2018**



## **ACTA DE SUSTENTACIÓN DEL GRADO DE MAESTRO EN INGENIERÍA**

En la ciudad universitaria de la esperanza, siendo las 15:30 pm horas del día martes 20 del mes de noviembre del año dos mil dieciocho, en el auditorio de la facultad de ingeniería, en cumplimiento a lo señalado en el reglamento de grados de maestría y doctorado de la Universidad de Huánuco, se reunió el jurado calificador integrando por los docentes:

- Mg. BERTHA LUCILA CAMPOS RÍOS
- Mg. JOHNNY PRUDENCIO JACHA ROJAS
- Mg. FRANK ERICK CÁMARA LLANOS

Nombrados mediante resolución N° 604-2018-D-EPG-UDH, para evaluar la tesis intitulada: **"APLICACIÓN DE PSP (PERSONAL SOFTWARE PROCESS) PARA EL DESARROLLO DEL SISTEMA DE BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO - 2016"**. Presentado por el Bach. SILVERIO BRAVO Fernando Fortunato para optar el grado de maestro en Ingeniería, mención en gerencia de sistemas y tecnologías de información.

Dicho acto de sustentación se desarrolla en dos etapas: exposición y absolución de preguntas procediéndose luego a la evaluación por parte de los miembros de jurado.

Habiéndose absuelto las objeciones que le fueron formuladas por los miembros del jurado y de conformidad con las respectivas disposiciones reglamentarias procedieron a deliberar y calificar, declarándolo APROBADO por UNANIMIDAD con calificativo cuantitativo de 16 y cualitativo de BUENO.

Siendo las 17:00 horas del día martes 20 del mes de noviembre del año dos mil dieciocho, los miembros del jurado calificador firman la presente acta en señal de conformidad.

Presidente

Mg. Bertha Lucila Campos Ríos

Secretario

Mg. Johnny Prudencio Jacha Rojas

Vocal

Mg. Frank Erick Cámara Llanos

## **DEDICATORIA**

A mis padres quienes con cariño y esfuerzo me condujeron en la vida, sin dudar en ningún instante de ver realizados mis sueños

# ÍNDICE

RESUMEN .....	6
ABSTRACT .....	7
INTRODUCCIÓN .....	8
CAPITULO I .....	9
PLANTEAMIENTO DE LA INVESTIGACIÓN .....	9
1.1. Descripción del problema.....	9
1.2. Formulación del problema .....	10
1.2.1. Problema General .....	10
1.2.2. Problemas Específicos .....	10
1.3. Objetivos generales .....	10
1.4. Objetivos específicos .....	11
1.5. Trascendencia y Justificación .....	11
1.6. Viabilidad de la Investigación .....	12
CAPITULO II .....	13
2. MARCO TEORICO.....	13
2.1. Antecedentes de la Investigación .....	13
2.1.1. Antecedente a nivel internacional: .....	13
2.1.2. Antecedente a nivel nacional:.....	14
2.1.3. Antecedente a nivel local:.....	14
2.2. Bases teóricas .....	15
PERSONAL SOFTWARE PROCESS (PSP).....	15
2.2.1. Definición de PSP .....	15
2.2.2. Elementos del PSP .....	15
2.2.3. Los Principios del PSP.....	16
2.2.4. La Estructura del Proceso PSP .....	17
2.2.5. Disciplina del PSP .....	18
2.2.6. Niveles del PSP .....	18
2.2.7. Fases del proceso en PSP .....	19
2.2.8. Los Estatutos del PSP y las Tendencias del Futuro .....	21
2.3. Definiciones Conceptuales .....	21
2.4. Sistema de Hipótesis .....	23
2.5. Sistema de Variables .....	23
2.6. Operacionalización de Variables.....	24
CAPITULO III .....	25
3. MARCO METODOLÓGICO.....	25
3.1. Tipo de investigación .....	25

3.2. Población y Muestra .....	25
3.3. Técnicas e instrumentos de recolección de datos .....	25
3.4. Técnicas para el procesamiento y análisis de la información. ....	25
CAPITULO IV .....	26
4. RESULTADOS .....	26
4.1. Descripción de la realidad observada .....	26
4.2. Aplicación del PSP (PROCESO DE SOFTWARE PERSONAL) .....	26
4.3. Análisis de datos del PSP (PERSONAL SOFTWARE PROCESS) .....	59
CAPITULO V .....	63
5. DISCUSIÓN DE RESULTADOS .....	63
6. CONCLUSIONES .....	68
7. RECOMENDACIONES .....	69
8. REFERENCIAS BIBLIOGRÁFICAS .....	70
9. ANEXOS .....	71
ANEXO 1: ESTÁNDARES DE CODIFICACIÓN .....	71
ANEXO 2: DOCUMENTOS PSP .....	72
ANEXO 3: SOFTWARE BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO .....	74
ANEXO 4: MATRIZ DE CONSISTENCIA .....	82
ANEXO 5: EVIDENCIAS DE APLICACIÓN Y DESARROLLO .....	83

## LISTA DE FIGURAS

Figura 1 Elementos del PSP .....	15
Figura 2 Flujo del proceso de PSP .....	17
Figura 3 Niveles del PSP .....	19
Figura 4 Fases del proceso en PSP .....	19

## **RESUMEN**

El principal objetivo del presente trabajo de investigación es Determinar en qué medida la aplicación de la metodología PSP mejorará el proceso de desarrollo del sistema de bolsa de trabajo de la Universidad de Huánuco.

La investigación es de tipo aplicativo y tecnológico porque utilizamos la teoría para desarrollar un producto software utilizando un nuevo enfoque en el proceso de desarrollo de software.

La investigación finaliza dando como resultado una metodología personal que servirá para el diseño individual de futuros proyectos de diseño de software.

Además muestra las conclusiones del uso del PSP y recomendaciones para la mejor adecuación a su aplicación con el fin de mejorar la calidad del producto de software a desarrollar.

**PALABRAS CLAVE:** PSP (Personal Software Process), Calidad de Software, Mejora Continua.

## **ABSTRACT**

The main objective of this research work is to determine to what extent the application of the PSP methodology will improve the process of developing the work exchange system of the University of Huánuco.

The research is of application and technological type because we use the theory to develop a software product using a new approach in the software development process.

The research ends with the result of a personal methodology that will be used for the individual design of future software design projects.

It also shows the conclusions of the use of the PSP and recommendations for the best adaptation to its application in order to improve the quality of the software product to be developed.

**KEY WORDS:** PSP (Personal Software Process), Software Quality, Continuous Improvement.



## INTRODUCCIÓN

Es muy conocido que en el desarrollo de software se dan inconvenientes que hacen que su tiempo de implementación se acreciente considerablemente en relación a lo previsto. Estos retrasos no solamente se presentan a nivel educativo sino en la industria del software, estos inconvenientes se ven manifestados en la demora de la entrega del sistema y también en la calidad del mismo.

Dada esta situación el presente trabajo de investigación trata de representar mediante un análisis en el proceso de desarrollo de software, cuáles son las causas que hacen que dichos sistemas o proyectos no puedan cumplir con su tiempo estimado de implementación y la calidad idónea para que el cliente quede satisfecho con el producto final. Para esto se utilizó la metodología PSP.

Así se desarrolla un procedimiento formal para la mejora en el desarrollo de un proyecto de software donde obtenemos valores medibles sobre las posibles fallas al momento de trabajar además de medir fortalezas y debilidades.

# CAPITULO I

## PLANTEAMIENTO DE LA INVESTIGACIÓN

### 1.1.Descripción del problema

El presente estudio explora el proceso de software personal (PSP) para luego aplicar en el desarrollo del software de bolsa de trabajo de la UDH.

La mayoría de modelos de procesos de desarrollo de software se ha desarrollado a nivel corporativo u organizacional para grandes equipos de desarrolladores. En nuestra región existe la necesidad de desarrollar software a pequeña escala como, que generalmente lo desarrolla una sola persona. Necesitamos un entorno ideal, crearía un proceso que mejor ajustara sus necesidades y, al mismo tiempo, satisfaga las necesidades más amplias del equipo y la organización. El propio equipo de desarrollo de software puede crear su propio proceso y, al mismo tiempo, satisfacer las necesidades más estrechas de los individuos y las necesidades más amplias de la organización. El mejor proceso de software es uno que está cerca de las personas que estarán haciendo el trabajo de ingeniería de software.

Según (*Humphrey, 1996*) el Proceso de software personal (PSP) proporciona a los ingenieros un marco personal disciplinado para hacer el trabajo de software. El proceso de PSP consiste en un conjunto de métodos, formularios y guiones que muestran a los ingenieros de software cómo planificar, medir y administrar su trabajo. El PSP está diseñado para ser utilizado con cualquier lenguaje de programación o metodología de diseño y puede ser utilizado para la mayoría de los aspectos del trabajo del software, incluyendo requisitos de escritura, pruebas en ejecución, definición de procesos y reparación de defectos. Cuando los ingenieros usan el PSP, el objetivo del proceso

recomendado es producir productos de defecto cero según lo programado y dentro de los costos planificados. Cuando se utiliza con el proceso de software de equipo (TSP), el PSP ha sido eficaz para ayudar a los ingenieros a alcanzar estos objetivos.

Esta investigación describe en detalle lo que es el PSP y cómo funciona. A partir de la relación entre el PSP y los principios generales de calidad, el trabajo de investigación describe cómo se desarrolló el PSP, sus principios y sus métodos. Además de la aplicación de las herramientas y documentación empleada en cada una de sus fases. El informe concluye dando como resultado una metodología personal que servirá para el diseño individual de futuros proyectos de diseño de software.

## **1.2. Formulación del problema**

### **1.2.1. Problema General**

¿En qué medida la aplicación de la metodología PSP mejora el proceso de desarrollo del sistema de bolsa de trabajo de la Universidad de Huánuco?

### **1.2.2. Problemas Específicos**

**P.E.1:** ¿Cómo beneficiará el uso de la metodología PSP al construir nuestro propio proceso de desarrollo?

**P.E.2:** ¿Cómo influenciará la aplicación de la metodología PSP en la optimización del desarrollo del software de bolsa de trabajo de la Universidad de Huánuco?

## **1.3. Objetivos generales**

Determinar en qué medida la aplicación de la metodología PSP mejorara el proceso de desarrollo del sistema de bolsa de trabajo de la Universidad de Huánuco.

#### **1.4. Objetivos específicos**

**O.E. 1:** Determinar el beneficio del uso de la metodología PSP al construir nuestro propio proceso de desarrollo.

**O.E. 2:** Determinar la influencia de la aplicación de la metodología PSP en la optimización del desarrollo del software de bolsa de trabajo de la Universidad de Huánuco.

#### **1.5. Trascendencia y Justificación**

Creemos que nuestro proyecto tendrá una gran trascendencia en la ingeniería de software; la trascendencia tecnológica del proyecto turismo es especialmente ostensible en el desarrollo de software de pequeña escala, software personales o de equipos de desarrollo de 2 a tres desarrolladores.

Es necesario obtener nuevas estrategias de desarrollo de software personal. El propio equipo de desarrollo de software puede crear su propio proceso y, al mismo tiempo, satisfacer las necesidades más estrechas de los individuos y las necesidades más amplias de la organización.

El mejor proceso de software es uno que está cerca de las personas que estarán haciendo el trabajo de ingeniería de software. La mayoría de procesos se ha desarrollado a nivel corporativo u organizacional. Necesitamos un entorno ideal, crearía un proceso que mejor se ajustare a sus necesidades y, al mismo tiempo, satisfaga las necesidades más amplias del equipo y la organización.

Los aspectos que alcanzaremos en la presente investigación y las Limitaciones exteriorizan qué aspectos permanecen fuera de su cobertura (las limitaciones de ningún modo se describen como las dificultades de realización, como muchos especulan, por lo contrario se representan con los límites o fronteras hasta donde alcanzan las ambiciones de la investigación, eso sí siempre con referencia a los objetivos).

## **1.6. Viabilidad de la Investigación**

Tomando en cuenta los recursos financieros, humanos y materiales de los que se pueden echar mano para realizar la investigación, hemos evaluado que estos son los suficientes para llevar a buen término la investigación.

## CAPITULO II

### MARCO TEORICO

#### 2.1. Antecedentes de la Investigación

Al realizar la revisión bibliográfica respecto al tema en las distintas instituciones públicas y privadas ubicamos las siguientes investigaciones.

##### 2.1.1. Antecedente a nivel internacional:

- a) Chávez Soledispa; Durán Hoyos (2015). “Aplicación de personal software process para el desarrollo del sistema administrable de códigos de barra a partir de la evaluación de procesos de reingeniería” tesis para obtener maestría en sistemas de información. Escuela Superior Politécnica del Litoral, Ecuador. El trabajo de investigación tiene los siguientes objetivos: Generar de manera automatizada el reporte de horas de ingreso y salidas por cada trabajador, realizar la revisión del pago por hora a cada empleado de acuerdo a su cargo, automatizar los pagos quincenales realizados por los empleados.
- b) Moreno Vallespir (2013). “pspvdc: propuesta de adaptacion del proceso de software personal para incorporar diseño por contrato verificado”. tesis para obtener maestría en informática. Universidad de la República. Uruguay. El trabajo de investigación tiene los siguientes objetivos: Conocer adaptaciones ya realizadas al PSP que incorporen métodos formales, construir una adaptación al PSP que incorpore VDbC, Planificar y diseñar un estudio empírico para comparar la calidad y productividad de PSPVDC con respecto a PSP.

### **2.1.2. Antecedente a nivel nacional:**

- a) Dioses Farfán (2015). “Desarrollo de la metodología para la implantar un sistema de gestión de calidad aplicado al software de computadora según la norma iso 9001:2000, con los lineamientos de la norma iso 90003:2004”. tesis para optar el título de ingeniero informático. Pontificia Universidad Católica DEL Perú. El trabajo de investigación tiene como objetivo concebir el diagnóstico del Sistema de Gestión de Calidad aplicado al área de desarrollo de software existente.
- b) Flores Jáuregui (2015). “construcción de un recurso cloud computing para facilitar la adopción del proceso personal de software en el desarrollo de software”. tesis para obtener el título profesional de ingeniero de computación y sistemas. Universidad Privada Antenor Orrego. El trabajo de investigación tiene como objetivo construcción de una solución Cloud Computing que automatiza las tareas del Proceso Personal de Software.

### **2.1.3. Antecedente a nivel local:**

A nivel local no se evidencio tesis semejantes debido a distintas razones detalladas a continuación:

- No existe un registro de tesis detallado.
- No se tomaron el tiempo en atender a la petición por averiguar la existencia de tesis.
- No existe la maestría en ingeniería con mención en gerencia de sistemas y tecnologías de información.

## 2.2. Bases teóricas

### PERSONAL SOFTWARE PROCESS (PSP)

#### 2.2.1. Definición de PSP

Carlos Javier Pérez Escobar (2015) afirma que: El PSP es un mejoramiento en sí de los procesos y fue diseñado para ayudar a los diseñadores de software a fiscalizar, administrar y optimizar su manera de trabajar. Es una estructura con un marco referencial, directrices y procedimientos para el desarrollo de software. El PSP suministra datos históricos para efectuar mejor su trabajo de acuerdo a los compromisos u obligaciones establecidos, haciendo que los elementos rutinarios de su trabajo sean más previsibles y más eficaces.

Además suministra métodos sumamente detallados de planificación y estimación, muestra a los ingenieros como controlar su rendimiento frente a dichos planes y también explica como los procesos definidos ordenan su trabajo.

#### 2.2.2. Elementos del PSP

El PSP se confecciona de 4 elementos fundamentales, que unidos aportan las herramientas para la mejora continua, estos elementos son:



Figura 1 Elementos del PSP

**Scripts:** Son los elementos que documentan el proceso, enseñan que hacer y cuando hacerlo. Siendo apegados a la definición formal, su



propósito es proveer una guía de alto nivel de cómo usar el proceso. Un ejemplo puede ser el siguiente (Script general del Proceso).

**Medidas:** Miden el proceso y el producto, muestran si las formas están desempeñando bien. Algunas de las medidas que PSP acopia se enfocan en 4 aspectos, Tamaño, Esfuerzo, Calidad y Programación (Agenda o Cronograma).

**Formas:** Son formularios para recopilar de manera sencilla y consistente la información. Entre los más básicos: Log de Tiempo (Donde se almacena cuando se invierte en cada fase o tarea del proyecto), Log de Defectos (En el cual se recopila la información de los defectos encontrados).

**Estándares:** Puntualizan como yo (personalmente) formo las cosas. Por ejemplo: Estándar de Código (Permite saber cómo cada uno de los desarrolladores escribirá su código).

### 2.2.3. Los Principios del PSP

Jesús Antonio Solís García (2012) afirma que: El diseño del PSP está basado en los siguientes principios de planificación y calidad:

- 1 Cada diseñador de software es diferente, para ser efectivo el diseñador de software debe planear su trabajo y basar sus planes de acuerdo a su información personal.
- 2 Para Mejorar la consistencia del desarrollo los diseñadores de software tienen que definir bien el uso y la medida del proceso.
- 3 Para producir productos de calidad los diseñadores de software deben sentirse personalmente responsables de la calidad del producto. Productos superiores no son producidos por error, el diseñador de software tiene que agotar todos los esfuerzos para llegar a trabajos de calidad.
- 4 Un defecto cuesta mucho menos detectarlo que tratar arreglarlo más tarde durante el proceso.

- 5 Es más eficaz prevenir los defectos que poder detectarlos y arreglarlos.
- 6 La forma correcta siempre es la forma más rápida y barata de hacer un adecuado trabajo.

El trabajo o labor de la ingeniería es forjar software correctamente, enseña que el diseñador de software tiene que planear el trabajo antes siquiera comprometerse a realizarlo.

Para entender su desempeño personal, ellos tienen que medir el tiempo que se van a demorar en cada etapa del trabajo, los defectos que van a inyectar y remover, y el tamaño del producto que van a producir.

Para producir productos de mayor calidad de manera consistente, los diseñadores de software deben planear, rastrear, y medir la calidad del producto, y además ellos deben enfocarse en la calidad desde el inicio del trabajo. Finalmente, ellos deben analizar todos los resultados de cada uno de sus trabajos y usar estos resultados para mejorar sus procesos personales.

#### 2.2.4. La Estructura del Proceso PSP

Moisés Salinas Silva (2013) afirma que: La estructura del proceso PSP inicia con los requerimientos, los cuales son el primer paso en el “proceso de planeación” del PSP. Existe una guía de planificación para este trabajo y una síntesis para poder registrar los datos de la planificación. Mientras los diseñadores de software van siguiendo la guía escrita para hacer el trabajo, registran el tiempo que tardaron.

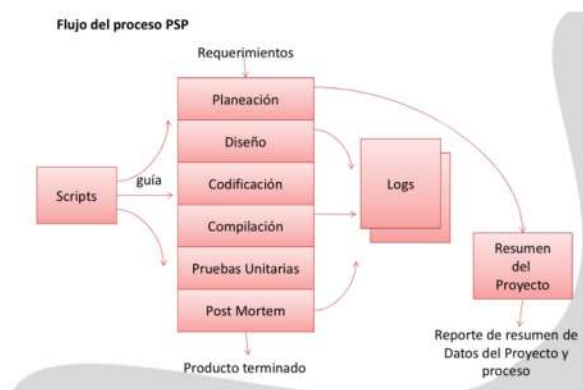


Figura 2 Flujo del proceso de PSP

### 2.2.5. Disciplina del PSP

Moisés Salinas Silva (2013) afirma que: En cualquier campo de la ingeniería el problema principal es conseguir que diseñador de software usen de forma permanente los métodos que ellos han aprendido. En el PSP, dichos métodos están alcanzando un proceso definido, planificando bien el trabajo, recogiendo los datos, y usando dichos datos para analizar y poder mejorar el proceso. Mientras esto parece simple en concepto, no es fácil en la práctica.

### 2.2.6. Niveles del PSP

Moisés Salinas Silva (2013) afirma que: PSP posee los siguientes niveles:

#### **PSP 0.**

**El proceso base** inculcar prácticas disciplinadas en el proceso de registro de datos

#### **PSO 1.**

**Proceso de planeación Personal** entender la relación entre tamaño de los programas y el tiempo que toma desarrollarlas.

#### **PSP 2.**

**Proceso de administración de calidad personal** manejar defectos de programación.

#### **PSP 3.**

**Proceso personal cíclico** para manejo de código grande (dividiéndolos en incrementos)

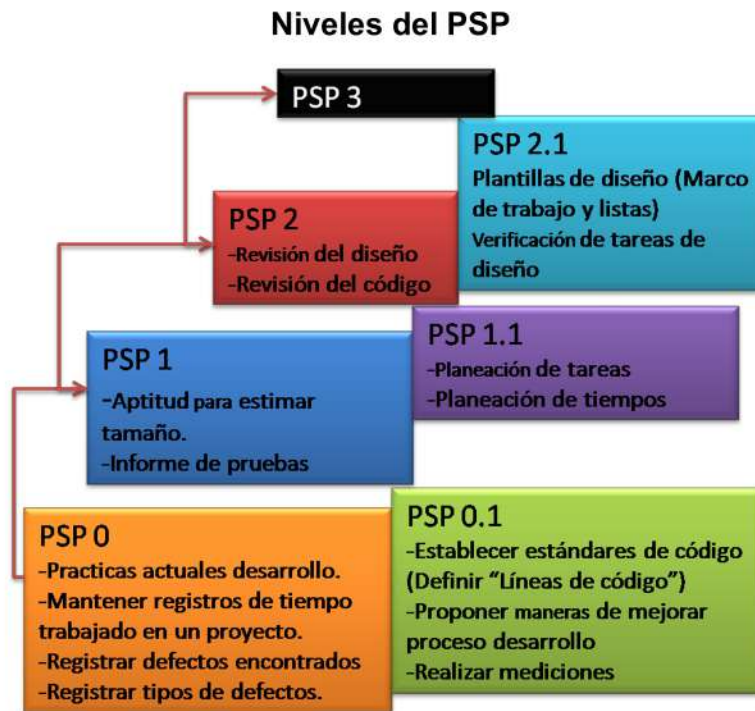


Figura 3 Niveles del PSP

## 2.2.7. Fases del proceso en PSP

### REQUISITOS

- Descripción del problema.
- Especificación de componentes.
- Formas de proceso.
- Estimadores del tamaño del producto y tiempos en base a históricos.



Figura 4 Fases del proceso en PSP

## **PLANEACIÓN (PLAN DE PROYECTO)**

**INPUT** Descripción detallada del problema, resumen del proyecto a detalle, el resumen es cíclico, tiempo estimado, tamaño estimado, formas de realizar planeación.

**ACTIVIDAD** Detalles de requerimientos, tamaños estimados, desarrollo de estrategia, estimación de recursos, programas y planificación de tareas, estimación de todos los defectos.

**OUTPUT** Diseño conceptual, resúmenes de planes, resumen de los ciclos, patrones estimados de planeación y tamaño de tareas, programas de patrones de planeación, registro de tiempos.

## **FASE DESARROLLO**

**INPUT** Diseño de alto nivel, tiempos y defectos, registro de seguimiento, ciclo de desarrollo, estrategia de pruebas, patrones de función y operación.

**ACTIVIDAD** Diseño de módulos, revisión de diseño, código, revisión de código, compilación, pruebas, aseguramiento de ciclo y de la calidad.

**OUTPUT** Patrón de diseño, módulos de software, lista de verificación de diseño y código, resumen de ciclo, patrón de reporte de pruebas, registro de tiempo, seguimiento y defectos.

## **FASE POSMORTEM**

**INPUT** Definición de requerimientos y problema, plan de ciclo y de proyecto, patrón de diseño, producto de software, lista de verificación de diseño y código, resumen del ciclo, patrón de reporte de pruebas, registro de tiempo, seguimiento y defectos.

**ACTIVIDAD** Defectos previstos, removidos, tamaño y tiempo del producto.

**OUTPUT** Productos, listas de verificación, plan de ciclo y proyecto, patrón de reporte de diseño y pruebas, forma con propuestas de mejora, registro seguimiento pruebas y tiempo.

### 2.2.8. Los Estatutos del PSP y las Tendencias del Futuro

Moisés Salinas Silva (2013) afirma que: En el futuro, los grupos de diseño de software, requerirán que se incremente la entrega de productos de calidad, a tiempo, y con costos planeados.

La Fiabilidad garantizada y nivel de servicio, seguridad de garantía y Penalidades en el contrato serán normales y el grupo de diseñadores de software que no puedan cumplir estas obligaciones no sobrevivirán.

Mientras el trabajo técnico continúe siendo requerido, la actuación individual de cada diseñador de software será reconocida como importante. Los sistemas de calidad requieren partes de calidad, y a menos que cada diseñador de software se esfuerce por producir trabajo de calidad, el equipo no puede funcionar. Dirección de calidad será una parte integral del entrenamiento del diseñador de software. Los diseñadores de software tendrán que aprender cómo medir la calidad de su trabajo y cómo usar estas medidas para producir trabajos libres de defecto.

### 2.3. Definiciones Conceptuales

- **LOC** Roberto Miñana (2016) afirma que: es un acrónimo de "Lines of Code". Se utiliza como métrica en diversas situaciones, en las que se mide el número de líneas de código.
- **Yield (Rendimiento del Proceso)** Roberto Miñana (2016) afirma que: es el porcentaje de defectos infectados y removidos antes de la primer compilación.
- **A/FR** Roberto Miñana (2016) afirma que: cociente entre costo de evaluación y costo de fallas
- **Modelo del Negocio (Modelo Conceptual)** • Jesús Antonio Solís García afirma que: el "modelo de negocio" es una técnica para comprender los procesos de negocios de la organización.
- **Requerimientos:** Jesús Antonio Solís García afirma que: los desarrolladores comienzan a planificar el trabajo que se va a realizar con todos los detalles posibles. Si lo que tienen es una

simple oración como informe, esta información debe ser la base del plan. Por supuesto, la exactitud de la estimación y plan, está relacionada directamente con el conocimiento que tenga el ingeniero acerca del trabajo que se va a realizar.

- **Diseño conceptual:** Jesús Antonio Solís García afirma que: para hacer una estimación y un plan, primero el desarrollador debe definir cómo va a diseñar el producto y construirlo. Sin embargo, en la fase de planificación se hace demasiado temprano para producir un diseño completo del producto, los desarrolladores producen lo que se llama un diseño conceptual. Durante la fase del diseño, los desarrolladores examinan alternativas del diseño y producen un diseño completo del producto.
- **Estimación del Tamaño del Producto y los Recursos:** Jesús Antonio Solís García afirma que: la correlación del tamaño del programa con tiempo de desarrollo, es moderado por buenos equipos de desarrolladores y organizaciones. Sin embargo, individualmente para los desarrolladores la correlación es habitualmente alta. Por consiguiente, el PSP inicia con la “estimación del tamaño” del producto que los desarrolladores desarrollaran personalmente. Entonces, basados en sus datos personales y en los datos de productividad, los ingenieros estiman el tiempo para hacer el trabajo.
- **El Horario de Producción:** Jesús Antonio Solís García afirma que: una vez que los desarrolladores saben el tiempo requerido para cada etapa del proceso, ellos, estiman el tiempo que ellos pasarán en el trabajo cada día o semana. Con esa información, ellos distribuyen el tiempo para realizar la tarea durante las horas establecidas, para producir en el tiempo planeado y completar la tarea.
- **Desarrollo del Producto:** Jesús Antonio Solís García afirma que: los desarrolladores realizan el trabajo programado. Mientras este trabajo normalmente no es considerado parte del proceso de la planificación, los desarrolladores deben utilizar estos datos del proceso para hacer planes futuros.

- **Análisis del Proceso:** Jesús Antonio Solís García afirma que: después de completar un trabajo, los desarrolladores hacen un análisis postmortem del trabajo. En el postmortem, ellos ponen al día el resumen del plan de proyecto con datos actuales, calculan cualquier cantidad requerida o desempeño de los datos, y la revisión que ellos realizaron contra el plan.

#### **2.4. Sistema de Hipótesis**

No aplicable, por estar relacionado al diseño de software y procesos de desarrollo del mismo.

#### **2.5. Sistema de Variables**

##### **Variable Independiente**

Aplicación del proceso de software personal

##### **Variable Dependiente**

Desarrollo del sistema de bolsa de trabajo



## 2.6. Operacionalización de Variables

VARIABLES	DIMENSIONES	INDICADORES
<b>VARIABLE INDEPENDIENTE</b> Aplicación del proceso de software personal (X)	Viabilidad del uso de la metodología PSP	beneficio del uso de la metodología PSP
		Razón de revisión
		Razón del tiempo de desarrollo
		Razón del defecto
		Productividad
<b>VARIABLE DEPENDIENTE</b> Desarrollo del sistema de bolsa de trabajo (Y)	Impacto en el desarrollo de software	optimización del desarrollo del software
		Defectos renovables viables
		Nivelación de la razón de falla

## **CAPITULO III**

### **MARCO METODOLÓGICO**

#### **3.1. Tipo de investigación**

Sánchez Carlessi H. y Reyes Meza C. (2006) afirman que: la investigación aplicada recibe el nombre de “investigación práctica o empírica”, que se caracteriza porque busca la aplicación o utilización de los conocimientos adquiridos, a la vez que se adquieren otros, después de implementar y sistematizar la práctica basada en investigación. El uso del conocimiento y los resultados de investigación que da como resultado una forma rigurosa, organizada y sistemática de conocer la realidad.

En ese sentido ya que la presente investigación aplica conocimientos adquiridos y además se generará otros conocimientos después de su implementación es de tipo aplicada.

#### **3.2. Población y Muestra**

No aplicable, por estar relacionado al tipo de investigación.

#### **3.3. Técnicas e instrumentos de recolección de datos**

Para realizar la recolección de datos y cumplir con los objetivos del presente trabajo de investigación se utilizara la herramienta “Process Dashboard” en su versión 2.4.

#### **3.4. Técnicas para el procesamiento y análisis de la información.**

Para el procesamiento y análisis de la información se utilizara la herramienta “Process Dashboard” en su versión 2.4.

## CAPITULO IV

### RESULTADOS

#### 4.1. Descripción de la realidad observada

En el desarrollo de software frecuentemente surgen las siguientes interrogantes: ¿mi software es de calidad?, ¿Esta mi cliente conforme con el producto?, ¿cumplo con mis proyectos en el tiempo estimado? Generalmente las respuestas a estas interrogantes son negativas; la mayor parte del software es de baja calidad, la mayoría de clientes solo son cliente por una vez y luego buscan nuevas alternativas, los proyectos se entregan fuera del tiempo estimado. Además estos problemas se acrecientan al realizar software de manera individual.

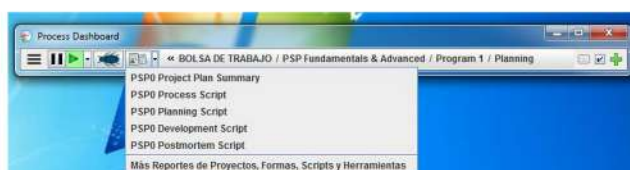
El presente trabajo de investigación pretende encaminar al desarrollador de software a dos metas fundamentales. El primero entregar el software a tiempo y el segundo entregar software de calidad para ello se aplica el Proceso de software personal (PSP) como alternativa de solución además se centra en la aplicación del PSP y se deja de lado las herramientas y metodologías utilizadas para el diseño del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO.

#### 4.2. Aplicación del PSP (PROCESO DE SOFTWARE PERSONAL)

Para aplicar el Proceso de software personal (PSP) se utilizara la herramienta “Process Dashboard” en su versión 2.4.

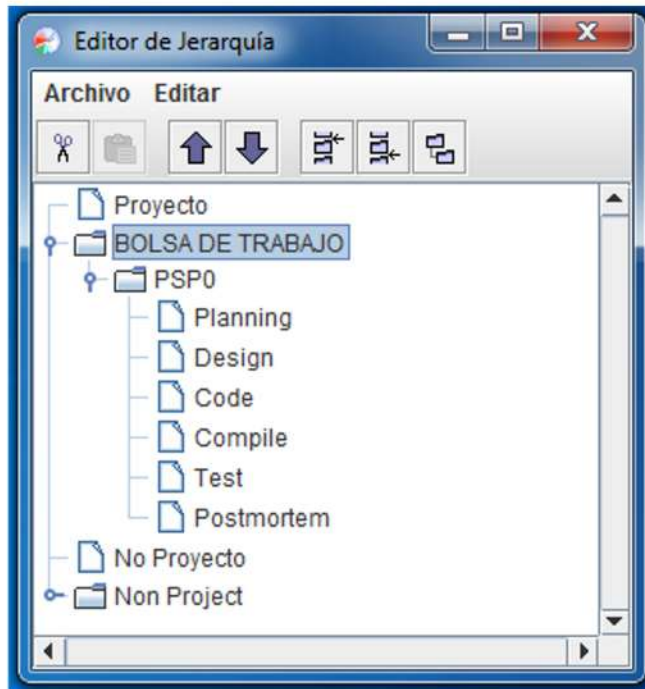


En la siguiente imagen se muestran las distintas herramientas para la aplicación del PSP.

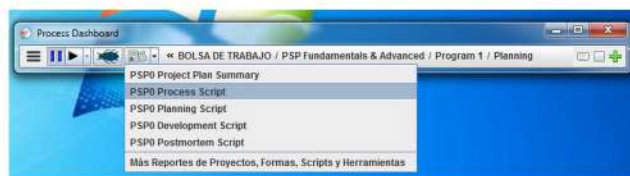


## INICIAR EL PSP0

Para iniciar con el nivel PSP0: el proceso base inculcar prácticas disciplinadas en el proceso de registro de datos la herramienta “Process Dashboard” nos permite gestionar las distintas categorías requeridas para completar dicho nivel.



Se elige la herramienta que muestra el SCRIPT DEL PROCESO de PSP



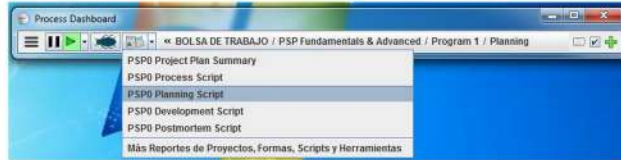
Se muestra el SCRIPT DEL PROCESO de PSP0

### Script de proceso PSP0

<b>Propósito</b>	Para guiar el desarrollo de programas a nivel de módulo
<b>Criterio para entrar</b>	<ul style="list-style-type: none"> <li>Descripción del problema</li> <li>PSP0 <a href="#">Resumen Plan de Proyecto</a> formulario</li> <li>Registros de registro de <a href="#">tiempo y defectos</a></li> <li><a href="#">Tipo de defecto estándar</a></li> <li><a href="#">Clasificación (opcional)</a></li> </ul>
<b>Paso/Ocupaciones</b>	<b>Descripción</b>
1 <a href="#">Planificación</a>	<ul style="list-style-type: none"> <li>Producir u obtener una <a href="#">declaración de requisitos</a></li> <li>Estime el tiempo de desarrollo requerido.</li> <li>Ingrese los datos del plan en el formulario <a href="#">Resumen</a> del plan del <a href="#">proyecto</a></li> <li>Complete el <a href="#">registro de registro de tiempo</a></li> </ul>
2 <a href="#">Desarrollo</a>	<ul style="list-style-type: none"> <li>Diseña el programa</li> <li>Implementa el diseño</li> <li>Compile el programa y corrija y registre todos los defectos encontrados</li> <li>Pruebe el programa y corrija y registre todos los defectos encontrados</li> <li>Complete el <a href="#">registro de registro de tiempo</a></li> </ul>
3 <a href="#">Post-mortem</a>	<ul style="list-style-type: none"> <li>Complete el formulario <a href="#">Resumen del plan del proyecto</a> con los datos de tiempo, defectos y <a href="#">tamaño</a> reales</li> </ul>
<b>Criterio de salida</b>	<ul style="list-style-type: none"> <li>Un programa completamente probado</li> <li>Formulario de <a href="#">resumen del plan de proyecto</a> completado con datos estimados y reales</li> <li>Registros de registro de <a href="#">tiempo y defectos</a> completados</li> </ul>

## Planificación

Se selecciona el SCRIPT DE PLANIFICACIÓN de la herramienta “Process Dashboard”



## Script de planificación PSP0

<b>Propósito</b>	Para guiar el proceso de planificación de PSP	
<b>Criterio para entrar</b>	<ul style="list-style-type: none"><li>• Descripción del problema</li><li>• Formulario de <a href="#">resumen del plan del proyecto</a></li><li>• <a href="#">Registro de tiempo de grabación</a></li></ul>	
<b>Paso</b>	<b>Ocupaciones</b>	<b>Descripción</b>
1	Requisitos del programa	<ul style="list-style-type: none"><li>• Producir u obtener una <a href="#">declaración de requisitos</a> para el programa.</li><li>• Asegúrese de que la declaración de requisitos sea clara y no ambigua.</li><li>• Resuelva cualquier pregunta.</li></ul>
2	Estimación de recursos	<ul style="list-style-type: none"><li>• Haga su mejor estimación del tiempo requerido para desarrollar este programa.</li><li>• Ingrese los datos de tiempo del plan en el formulario <a href="#">Resumen del plan del proyecto</a>.</li></ul>
<b>Criterio de salida</b>	<ul style="list-style-type: none"><li>• <a href="#">Declaración de requisitos</a> documentados</li><li>• Formulario de <a href="#">resumen del plan de proyecto</a> completado con datos de tiempo de desarrollo estimados</li><li>• <a href="#">Registro de registro de tiempo</a> completado</li></ul>	

[Siguiente: Desarrollo](#).

[Parte superior](#)

Para continuar con la planificación del proyecto se requieren de 2 pasos:

### Paso 1

Se debe tener claro los requerimientos para el proyecto en este caso se definieron los requerimientos para el diseño del software bolsa de trabajo de la universidad de Huánuco.

### Paso 2

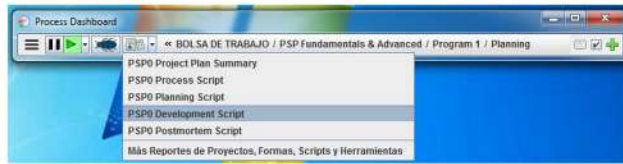
Se ingresa el RESUMEN DEL PLAN DEL PROYECTO

Resumen del plan de proyecto PSP0

Duño del proyecto	FERNANDO SILVERIO BRAVO			
Fecha de inicio				
Fecha de Terminación	Terminado: <input type="checkbox"/>			
Palabra clave				
Idioma	?????			
<b>Tiempo en fase (min)</b>	<b>Plan</b>	<b>Real</b>	<b>Hasta la fecha</b>	<b>Hasta la fecha %</b>
Planificación	0:00	0:00	0:00	100%
Diseño	0:00	0:00	0:00	100%
Código	0:00	0:00	0:00	100%
Compilar	0:00	0:00	0:00	100%
Prueba	0:00	0:00	0:00	100%
Post mortem	0:00	0:00	0:00	100%
<b>Total</b>	<b>0:00</b>	<b>0:00</b>	<b>0:00</b>	
<b>Defectos inyectados</b>	<b>Real</b>	<b>Hasta la fecha</b>	<b>Hasta la fecha %</b>	
Antes del desarrollo	0	0	100%	
Planificación	0	0	100%	
Diseño	0	0	100%	
Código	0	0	100%	
Compilar	0	0	100%	
Prueba	0	0	100%	
Desarrollo total	0	0	100%	
<b>Defectos eliminados</b>	<b>Real</b>	<b>Hasta la fecha</b>	<b>Hasta la fecha %</b>	
Planificación	0	0	100%	
Diseño	0	0	100%	
Código	0	0	100%	
Compilar	0	0	100%	
Prueba	0	0	100%	
Desarrollo total	0	0	100%	
Después del desarrollo	0	0	100%	

## Desarrollo

Se selecciona el SCRIPT DE DESARROLLO de la herramienta “Process Dashboard”



## Script de desarrollo PSP0

<b>Propósito</b>	Para guiar el desarrollo de pequeños programas	
<b>Criterio para entrar</b>	<ul style="list-style-type: none"> <li>Declaración de requisitos</li> <li>Formulario de resumen del plan del proyecto con tiempo estimado de desarrollo del programa</li> <li>Registros de registro de tiempo y defectos</li> <li>Tipo de defecto estándar</li> </ul>	
<b>Paso</b>	<b>Ocupaciones</b>	<b>Descripción</b>
1	Diseño	<ul style="list-style-type: none"> <li>Revise los requisitos y produzca un diseño para cumplirlos.</li> <li>Registre en el registro de registro de defectos cualquier defecto encontrado en los requisitos.</li> <li>Tiempo de registro en el registro de tiempo de grabación.</li> </ul>
2	Código	<ul style="list-style-type: none"> <li>Implementa el diseño.</li> <li>Registre en el registro de grabación de defectos cualquier requisito o defecto de diseño encontrado.</li> <li>Tiempo de registro en el registro de tiempo de grabación.</li> </ul>
3	Compilar	<ul style="list-style-type: none"> <li>Compile el programa hasta que no haya errores de compilación.</li> <li>Reparar todos los defectos encontrados.</li> <li>Grabar defectos en el registro de registro de defectos.</li> <li>Tiempo de registro en el registro de tiempo de grabación.</li> </ul>
4	Prueba	<ul style="list-style-type: none"> <li>Pruebe hasta que todas las pruebas se ejecuten sin error.</li> <li>Reparar todos los defectos encontrados.</li> <li>Grabar defectos en el registro de registro de defectos.</li> <li>Tiempo de registro en el registro de tiempo de grabación.</li> </ul>
<b>Criterio de salida</b>	<ul style="list-style-type: none"> <li>Un programa completamente probado</li> <li>Registros de registro de tiempo y defectos completados</li> </ul>	

[Siguiendo Postmortem](#)

[Parte superior](#)

## Diseño

Se revisan los requisitos del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO.

Se registra los tiempos del diseño en la herramienta “Process Dashboard”



### Código

Se ingresan los errores presentados en la etapa de codificación en la herramienta “Process Dashboard”



Se ingresa los errores presentados en la etapa de codificación estos pueden ser de documentación, interface, base de datos, funciones o procedimientos internos, etc.



## Compilar

Se mide el tiempo de compilación del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO en la herramienta “Process Dashboard”



## Prueba

Se mide el tiempo de pruebas del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO en la herramienta “Process Dashboard”

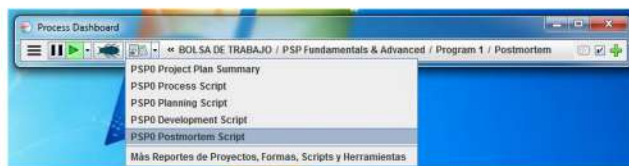


## Post Mortem

Se inicia la etapa de POST MORTEM en la herramienta “Process Dashboard”



Elegimos el SCRIPT DE POST MORTEM





## PSP0 Postmortem Script

<b>Propósito</b>	Para guiar el proceso post mortem de PSP
<b>Criterio para entrar</b>	<ul style="list-style-type: none"> <li>Descripción del problema y <a href="#">declaración de requisitos</a></li> <li>Formulario de <a href="#">resumen del plan del proyecto</a> con datos de tiempo de desarrollo</li> <li>Registros de registro de <a href="#">tiempo</a> y <a href="#">defectos</a> completados</li> <li>Un programa probado y en ejecución</li> </ul>
<b>Paso</b>	<b>Ocupaciones</b>
1	Grabación de defectos
2	Consistencia de datos de defecto
3	Hora
<b>Descripción</b>	<ul style="list-style-type: none"> <li>Revise el <a href="#">resumen del plan del proyecto</a> para verificar que se hayan registrado todos los defectos encontrados en cada fase.</li> <li>Usando su mejor recuerdo, registre cualquier defecto omitido.</li> <li>Verifique que los datos de cada defecto en el <a href="#">registro de registro de defectos</a> sean precisos y completos.</li> <li>Verifique que el número de defectos inyectados y eliminados por fase sea razonable y correcto.</li> <li>Con su mejor recuerdo, corrija los datos de defectos faltantes o incorrectos.</li> <li>Revise el <a href="#">registro de registro de tiempo</a> completo para ver si hay errores u omisiones.</li> <li>Usando su mejor recuerdo, corrija cualquier dato de tiempo faltante o incompleto.</li> </ul>
<b>Criterio de salida</b>	<ul style="list-style-type: none"> <li>Un programa completamente probado</li> <li>Formulario de <a href="#">resumen del plan de proyecto</a> completado</li> <li>Registros de registro de <a href="#">tiempo</a> y <a href="#">defectos</a> completados</li> </ul>

[Parte superior](#)

## Bitácora de tiempo

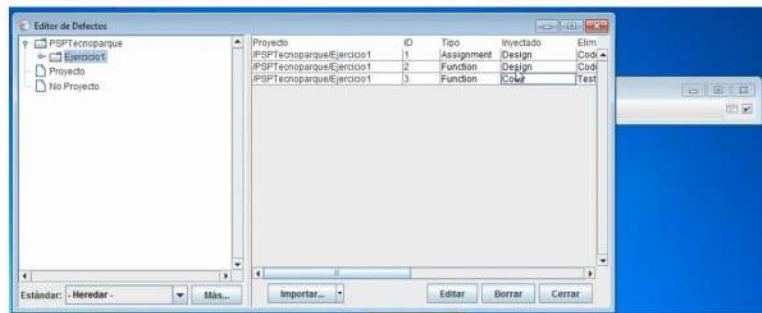
Se utiliza la herramienta “Process Dashboard” para verificar el registro de los tiempos en las distintas etapas realizadas previamente.



## Bitácora de defectos

Se utiliza la herramienta “Process Dashboard” para verificar el registro de los defectos encontrados en las distintas etapas realizadas previamente





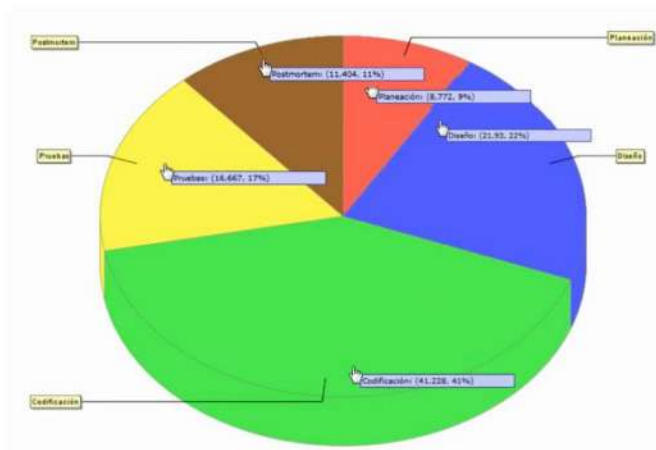
## Análisis de PSP 0

La herramienta "Process Dashboard" nos muestra los datos recopilados



## Análisis de la planeación

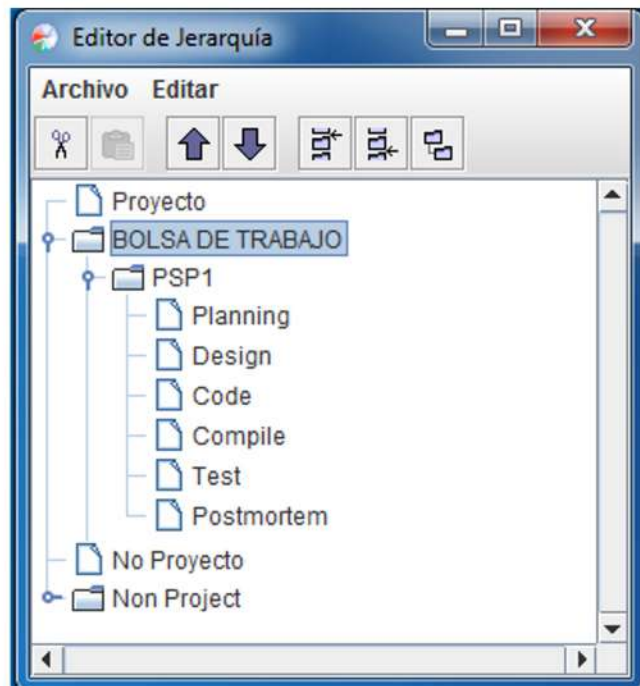




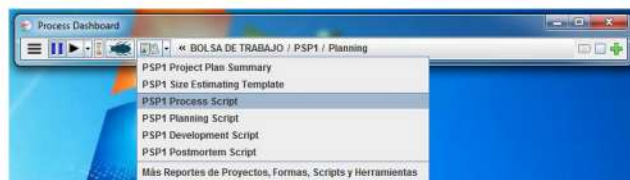
Estas métricas obtenidas en PSP 0 se utilizan para mejorar la calidad del proceso en PSP 0.1 a través del establecimiento de estándares de codificación orientados a mejorar las estimaciones de tamaño y revisión de errores presentados.

## INICIAR EL PSP1

Para iniciar con el nivel PSP1: Proceso de planeación Personal entender la relación entre tamaño de los programas y el tiempo que toma desarrollarlas la herramienta “Process Dashboard” nos permite gestionar las distintas categorías requeridas para completar dicho nivel.



Se elige la herramienta que muestra el SCRIPT DEL PROCESO de PSP



Se muestra el SCRIPT DEL PROCESO de PSP1

## Script de proceso PSP1

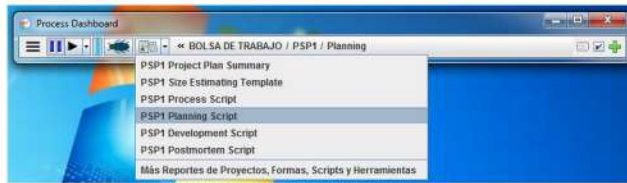
<b>Propósito</b>	Para guiar el desarrollo de programas a nivel de módulo
<b>Criterio para entrar</b>	<ul style="list-style-type: none"> <li>Descripción del problema</li> <li>PSP1 <b>Resumen Plan de Proyecto</b> formulario</li> <li><b>Plantilla de estimación de tamaño</b></li> <li><b>Datos históricos de tamaño y tiempo (estimados y reales)</b></li> <li>Registros de registro de <b>tiempo y defectos</b></li> <li><b>Tipo de defecto</b> , <b>codificación</b> y estándares de conteo de tamaño</li> <li>Cronograma (opcional)</li> </ul>
<b>Paso/Ocupaciones</b>	<b>Descripción</b>
1	<b>Planificación</b>
	<ul style="list-style-type: none"> <li>Producir u obtener una <b>declaración de requisitos</b>.</li> <li>Use el <b>método PROBE</b> para estimar el tamaño agregado y modificado de este programa.</li> <li>Complete la <b>plantilla de Estimación del tamaño</b>.</li> <li>Use el <b>método PROBE</b> para estimar el tiempo de desarrollo requerido.</li> <li>Ingrese los datos del plan en el formulario <b>Resumen</b> del plan del <b>proyecto</b>.</li> <li>Complete el <b>registro de registro de tiempo</b>.</li> </ul>
2	<b>Desarrollo</b>
	<ul style="list-style-type: none"> <li>Diseña el programa.</li> <li>Implementa el diseño.</li> <li>Compile el programa y corrija y registre todos los defectos encontrados.</li> <li>Pruebe el programa y corrija y registre todos los defectos encontrados.</li> <li>Complete el <b>registro de registro de tiempo</b>.</li> </ul>
3	<b>Post mortem</b>
	Complete el formulario de <b>Estimación del tamaño y el Resumen del plan del proyecto</b> con los datos de <b>tiempo, defectos y tamaño reales</b> .
<b>Criterio de salida</b>	<ul style="list-style-type: none"> <li>Un programa completamente probado</li> <li>Formulario de <b>resumen del plan de proyecto</b> completado con datos estimados y reales</li> <li>Completado <b>plantilla de la estimación del tamaño</b></li> <li>Completado <b>la plantilla Informe de prueba</b></li> <li>Formularios PIP completados</li> <li>Registros de registro de <b>tiempo y defectos</b> completados</li> </ul>

Es necesario un ESTÁNDAR DE CODIFICACIÓN (ANEXO 1) que consiste en la forma de escribir código propia de cada programador y completamente diferente a la forma de cualquier otro. De la forma usada depende la facilidad para entender el código y retomar ciertas partes realizadas por otros integrantes, así como la depuración de las mismas.

ESTANDAR DE CODIFICACION	
<b>Propósito</b>	Este documento tiene como propósito guiar la implementación de los programas que desarrollare utilizando el lenguaje de programación Java
<b>Conteo Estándar</b>	<ul style="list-style-type: none"> <li>Cada línea física representa una línea de código.</li> <li>No se cuentan líneas que solo contienen espacios en blanco ni líneas correspondientes a documentación.</li> <li>No se cuentan paquetes.</li> <li>No se tendrá en cuenta líneas autogeneradas por el entorno.</li> </ul>
<b>Campos, variables.</b>	<ul style="list-style-type: none"> <li>Los nombres de campos y variables iniciaran con letra minúscula.</li> <li>Si tienen nombre compuesto este debe ir sin espacios y la palabra que lo compone debe comenzar con caracteres en mayúscula.</li> <li>Los nombres no deben contener caracteres símbolos.</li> <li>No deben empezar por carácter número.</li> <li>Los nombres deben contener como mínimo 3 caracteres y estos deben dar información de lo que contienen el campo.</li> </ul>
<b>Interfaces, estructuras, clases</b>	<ul style="list-style-type: none"> <li>Deben ajustarse a las reglas de nombres definidos para las variables con la diferencia que los nombres de estos deben iniciar con letra mayúscula.</li> <li>El nombre debe expresar de manera breve la funcionalidad de la clase.</li> <li>Tendrá un encabezado distintivo "Class" para la clase. Y después de la línea que cierra la clase tendrá una línea de documentación "//SeoClass" para identificar el final de la clase.</li> <li>Tendrá un encabezado distintivo "Estruct" para la Estructura.</li> <li>Tendrá un encabezado distintivo "Interf" para la interfaz.</li> </ul>
<b>Bloques de código</b>	Deben tener una sangría de mínimo dos espacios para facilitar la interpretación del código en las líneas que conforman un bloque.
<b>Comentarios</b>	
<b>De Línea</b>	<ul style="list-style-type: none"> <li>Se utilizará una línea por comentario y esta podrá o no finalizar en punto.</li> <li>Se debe prestar atención a la ortografía.</li> <li>Se usaran para describir el funcionamiento de variables, métodos y clases.</li> </ul>
<b>De bloque</b>	<ul style="list-style-type: none"> <li>Iniciará con /*.</li> <li>En los intermedios iniciará con *.</li> <li>Terminará con */.</li> <li>Se utilizará una línea por comentario y esta podrá o no finalizar con punto.</li> <li>Se utilizará para el encabezado el cual contendrá nombre del programa, fecha, nombre de quien lo realiza y una breve descripción.</li> </ul>
<b>Métodos o funciones</b>	<ul style="list-style-type: none"> <li>Los nombres se ajustaran a las reglas definidas para nombrar clases.</li> <li>El nombre debe dar una breve descripción de lo que hace.</li> <li>Tendrá un encabezado distintivo "Método" para la clase.</li> <li>Tendrá una línea de comentario en la línea siguiente de la línea de cierre para identificación "//SeoMétodo".</li> </ul>
<b>Sentencias de decisión</b>	<ul style="list-style-type: none"> <li>Utilizará líneas aunque solo contenga una línea.</li> <li>Utilizar una sintaxis explicativa para los campos que están siendo objeto de comparación.</li> </ul>
<b>Sentencias de iteración</b>	<ul style="list-style-type: none"> <li>Documentar las sentencias que generan ruptura.</li> </ul>

## Planificación

Se selecciona el SCRIPT DE PLANIFICACIÓN de la herramienta “Process Dashboard”



### PSP1 Planning Script

<b>Propósito</b>	Para guiar el proceso de planificación de PSP	
<b>Criterio para entrar</b>	<ul style="list-style-type: none"> <li>Descripción del problema</li> <li>PSP1 Resumen Plan de Proyecto formulario</li> <li>Plantilla de estimación de tamaño</li> <li>Datos históricos de tamaño y tiempo (estimados y reales)</li> <li>Registro de tiempo de grabación</li> </ul>	
<b>Paso</b>	<b>Ocupaciones</b>	<b>Descripción</b>
1	Requisitos del programa	<ul style="list-style-type: none"> <li>Producir u obtener una <a href="#">declaración de requisitos</a> para el programa.</li> <li>Asegúrese de que la declaración de requisitos sea clara y no ambigua.</li> <li>Resuelva cualquier pregunta.</li> </ul>
2	Estimación del tamaño	<ul style="list-style-type: none"> <li>Producir un programa de <a href="#">diseño conceptual</a>.</li> <li>Use el <a href="#">método PROBE</a> para estimar el tamaño agregado y modificado de este programa.</li> <li>Complete la <a href="#">plantilla de Estimación del tamaño</a> y el formulario <a href="#">Resumen del plan del proyecto</a>.</li> </ul>
3	Estimación de recursos	<ul style="list-style-type: none"> <li>Use el <a href="#">método PROBE</a> para estimar el tiempo requerido para desarrollar este programa.</li> <li>Usando el% hasta la fecha del último programa desarrollado como guía, el tablero distribuirá automáticamente el tiempo de desarrollo sobre las fases planificadas del proyecto.</li> </ul>
<b>Criterio de salida</b>	<ul style="list-style-type: none"> <li><a href="#">Declaración de requisitos</a> documentados</li> <li><a href="#">Diseño conceptual del programa</a></li> <li><a href="#">Completado plantilla de la estimación del tamaño</a></li> <li>Formulario de <a href="#">resumen del plan del proyecto</a> completado con el tamaño estimado del programa y los datos de tiempo de desarrollo</li> <li><a href="#">Registro de registro de tiempo</a> completado</li> </ul>	

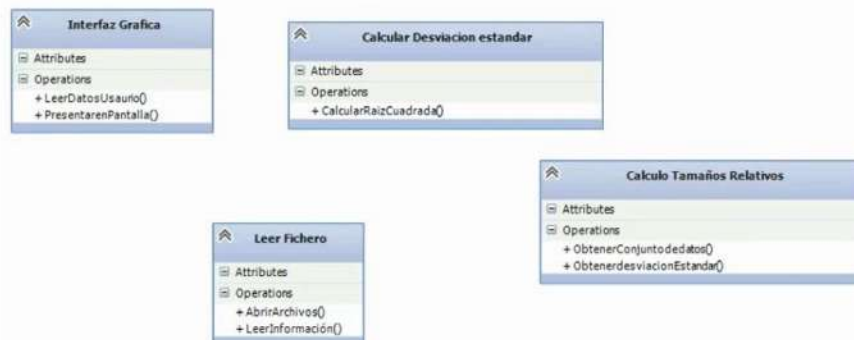
Siguiente: [Desarrollo](#).

[Parte superior](#)

Para realizar la estimación del tamaño se requiere de:

1. Diseño conceptual
2. Método PROBE

1. Diseño conceptual utilizada para el desarrollo del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO.



## 2. método PROBE

### Desde el SCRIPT DE PLANIFICACIÓN se accede al método PROBE

#### PSP1 Planning Script

3	Estimación de recursos	<ul style="list-style-type: none"> <li>Use el método PROBE para estimar el tiempo requerido para desarrollar este programa.</li> <li>Usando el% hasta la fecha del último programa desarrollado como guía, el tablero distribuirá automáticamente el tiempo de desarrollo sobre las fases planificadas del proyecto.</li> </ul>
---	------------------------	---

#### PROBE Estimating Script

Propósito	Para guiar el proceso de estimación de tamaño y tiempo utilizando el método PROBE
<b>Criterio para entrar</b>	<ul style="list-style-type: none"> <li>Declaración de requisitos</li> <li>Tamaño Estimación de plantilla e instrucciones</li> <li>Tamaño por datos de actividad para tipos de piezas</li> <li>Registro de tiempo de grabación</li> <li>Datos históricos de tamaño y tiempo</li> </ul>
<b>General</b>	<ul style="list-style-type: none"> <li>Este script asume que está usando datos de tamaño agregados y modificados como los tipos de contabilidad de tamaño para hacer estimaciones de tamaño y tiempo.</li> <li>Si elige algunos otros tipos de contabilidad de tamaño, reemplace cada "agregado y modificado" en este script con los tipos de contabilidad de tamaño de su elección.</li> </ul>
<b>Pase/Ocupaciones</b>	<b>Descripción</b>
1 Diseño conceptual	<ul style="list-style-type: none"> <li>Revisar los requisitos y producir un <a href="#">diseño conceptual</a></li> </ul>
2 Adiciones de partes	<ul style="list-style-type: none"> <li>Signa las <a href="#">instrucciones de la plantilla de estimación de tamaño</a> para estimar las adiciones de partes y los nuevos tamaños de piezas reutilizables.</li> </ul>
3 Piezas de base y piezas reutilizadas	<ul style="list-style-type: none"> <li>Para el programa base, calcule el tamaño del código base, eliminado, modificado y agregado.</li> <li>Mida y / o estime el tamaño de las piezas que se reutilizarán.</li> </ul>
<p><i>Nota: Los pasos a continuación pueden ser realizados automáticamente por el <a href="#">PROBE Wizard</a>. Por lo tanto, si está utilizando el asistente (recomendado), no es necesario continuar con el resto de este script.</i></p>	
4 Procedimiento de estimación de tamaño	<ul style="list-style-type: none"> <li>Si tiene un tamaño de proxy estimado suficiente y datos de tamaño real agregados y modificados (tres o más puntos que se correlacionan), utilice el procedimiento 4A.</li> <li>Si no tiene suficientes datos estimados pero tiene suficiente plan agregado y modificado y datos de tamaño agregados y modificados reales (tres o más puntos que se correlacionan), utilice el procedimiento 4B.</li> <li>Si no tiene datos históricos, use el procedimiento 4C.</li> </ul>
4A Procedimiento de estimación de tamaño 4A	<ul style="list-style-type: none"> <li>Utilizando el método de regresión lineal, calcule los parámetros <math>B_0</math> y <math>B_1</math> partir del tamaño de proxy estimado y los datos de tamaño real agregados y modificados.</li> <li>Si el valor absoluto de <math>B_1</math> no es cercano a 0 (menos de aproximadamente el 25% del tamaño esperado del nuevo programa), o <math>B_0</math> no está cerca de 1.0 (entre aproximadamente 0.5 y 2.0), utilice el procedimiento 4B.</li> </ul>
4B Procedimiento de estimación de tamaño 4B	<ul style="list-style-type: none"> <li>Utilizando el método de regresión lineal, calcule los parámetros <math>B_0</math> y <math>B_1</math> partir del plan agregado y el tamaño modificado y los datos reales agregados y modificados.</li> <li>Si el valor absoluto de <math>B_1</math> no es cercano a 0 (menos de aproximadamente el 25% del tamaño esperado del nuevo programa), o <math>B_0</math> no está cerca de 1.0 (entre aproximadamente 0.5 y 2.0), utilice el procedimiento 4C.</li> </ul>
4C Procedimiento de estimación de tamaño 4C	<ol style="list-style-type: none"> <li>Si tiene datos sobre el tamaño del proxy del plan y el tamaño real agregado y modificado, establezca <math>B_0 = 0</math> y <math>B_1 =</math> (tamaño real agregado y modificado actual hasta la fecha / planifique el tamaño total del proxy hasta la fecha).</li> <li>Si tiene datos sobre el plan agregado y el tamaño modificado y el tamaño real agregado y modificado, establezca <math>B_0 = 0</math> y <math>B_1 =</math> (total agregado y modificado actual hasta la fecha / plan agregado total y tamaño modificado hasta la fecha).</li> </ol>
4D Procedimiento de estimación de tamaño 4D	<ul style="list-style-type: none"> <li>Si no tiene datos históricos, use su juicio para estimar el tamaño agregado y modificado.</li> </ul>
5 Procedimiento de estimación del tiempo	<ul style="list-style-type: none"> <li>Si tiene suficiente tamaño de proxy estimado y datos de tiempo de desarrollo reales (tres o más puntos que se correlacionan), utilice el procedimiento 5A.</li> <li>Si no tiene suficientes datos estimados de tamaño pero tiene suficiente plan agregado y modificado de tamaño y datos de tiempo de desarrollo real (tres o más puntos que se correlacionan), utilice el procedimiento 5B.</li> <li>Si tiene datos insuficientes o no se correlacionan, use el procedimiento 5C.</li> <li>Si no tiene datos históricos, use el procedimiento 5D.</li> </ul>
5A Procedimiento de estimación de tiempo 5A	<ul style="list-style-type: none"> <li>Utilizando el método de regresión lineal, calcule los parámetros <math>B_0</math> y <math>B_1</math> partir del tamaño de proxy estimado y los datos de tiempo de desarrollo total real.</li> <li>Si <math>B_1</math> no está cerca de 0 (estrictamente más pequeño que el tiempo de desarrollo esperado para el nuevo programa), o <math>B_0</math> no está dentro del 50% de 1 (productividad histórica), utilice el procedimiento 5B.</li> </ul>
5B Procedimiento de estimación de tiempo 5B	<ul style="list-style-type: none"> <li>Usando el método de regresión lineal, calcule los parámetros de regresión <math>B_0</math> y <math>B_1</math> partir del plan agregado y el tamaño modificado y los datos del tiempo de desarrollo total real.</li> <li>Si <math>B_1</math> no está cerca de 0 (estrictamente más pequeño que el tiempo de desarrollo esperado para el nuevo programa), o <math>B_0</math> no está dentro del 50% de 1 (productividad histórica), utilice el procedimiento 5C.</li> </ul>
5C Procedimiento de estimación de tiempo 5C	<ol style="list-style-type: none"> <li>Si tiene datos sobre el tamaño del proxy del plan y el tiempo real de desarrollo, configure <math>B_0 = 0</math> y <math>B_1 =</math> (tiempo de desarrollo total real hasta la fecha / planifique el tamaño del proxy total hasta la fecha).</li> <li>Si tiene datos sobre el plan agregado y el tamaño modificado y el tiempo de desarrollo real, configure <math>B_0 = 0</math> y <math>B_1 =</math> (tiempo de desarrollo total actual hasta la fecha / planifique el tamaño agregado y modificado total hasta la fecha).</li> <li>Si solo tiene datos de tiempo y tamaño reales, configure <math>B_0 = 0</math> y <math>B_1 =</math> (tiempo total real de desarrollo hasta la fecha / total actual agregado y tamaño modificado hasta la fecha).</li> </ol>
5D Procedimiento de estimación de tiempo 5D	<ul style="list-style-type: none"> <li>Si no tiene datos históricos, use su juicio para estimar el tiempo de desarrollo del tamaño estimado agregado y modificado.</li> </ul>
6 Intervalos de predicción de tiempo y tamaño	<ul style="list-style-type: none"> <li>Si usó el método de regresión A o B, calcule los intervalos de predicción del 70% para las estimaciones de tiempo y tamaño.</li> <li>Si no usó el método de regresión o no sabe cómo calcular el intervalo de predicción, calcule los límites de estimación de tiempo de desarrollo mínimo y máximo de su productividad histórica máxima y mínima para los programas escritos hasta la fecha.</li> </ul>
<b>Criterio de validación</b>	<ul style="list-style-type: none"> <li>Entradas estimadas y reales completadas para todas las categorías de tamaño pertinentes</li> <li>Se completó la <a href="#">Hoja de cálculo de PROBE</a> con entradas de tamaño y tiempo</li> <li>Plan y valores reales ingresados en el <a href="#">Resumen del Plan del Proyecto</a></li> </ul>

Siguiente: [Continuar la planificación](#)

[Parte superior](#)

## Aplicación del método PROBE

### Desde el SCRIPT DE PLANIFICACIÓN se accede a la plantilla de estimación de tamaño

#### PSP1 Planning Script

2	Estimación del tamaño	<ul style="list-style-type: none"> <li>Resuelva cualquier pregunta.</li> <li>Producir un programa de <a href="#">diseño conceptual</a>.</li> <li>Use el método PROBE para estimar el tamaño agregado y modificado de este programa.</li> <li>Complete la <a href="#">plantilla de Estimación del tamaño</a> y el <a href="#">formulario Resumen del plan del proyecto</a>.</li> </ul>
3	Estimación de recursos	<ul style="list-style-type: none"> <li>Use el método PROBE para estimar el tiempo requerido para desarrollar este programa.</li> <li>Usando el% hasta la fecha del último programa desarrollado como guía, el tablero distribuirá automáticamente el tiempo de desarrollo sobre las fases planificadas del proyecto.</li> </ul>

## Completamos la PLANTILLA DE ESTIMACIÓN DE TAMAÑO

### Plantilla de estimación de tamaño [\(instrucciones...\)](#)

Dueño del proyecto: FERNANDO SILVERIO BRAVO  
 Medida de tamaño: LOC

PIEZAS BASE	Estimado				Real			
	BASE	ELIMINADO	MODIFICADO	ADICIONAL	BASE	ELIMINADO	MODIFICADO	ADICIONAL
Modulo Alumno	1000	100	50	200	0	0	0	0
<b>agregue más filas para partes de base</b>	<b>Total:</b>	<b>1000</b>	<b>100</b>	<b>50</b>	<b>200</b>	<b>0</b>	<b>0</b>	<b>0</b>

ADICIONES DE PIEZAS	Estimado				Real			
	TPO	ARTICULOS	REL. TAMAÑO	TAMAÑO	TAMAÑO	ARTICULOS	REL.	REL.
Clase Desviación Estandar	Cálculo	1	Medio	11,3				
Leer Ficheros	E / S	2	Pequeña	24,1				
Calculo Tamaños Relativos	Cálculo	2	Grande	49,3				
<b>agregue más filas para agregar partes...</b>	<b>Total:</b>			<b>84,7</b>	<b>0</b>			

PIEZAS REUTILIZADAS	Estimado	Real
	TAMAÑO	TAMAÑO
Clase IntervaloProduccion	67	0
<b>agregue más filas para piezas reutilizadas...</b>	<b>Total:</b>	<b>67</b>


TAMAÑO TOTAL	Real
Tamaño real del producto terminado:	TAMAÑO

Utilizamos el ASISTENTE PROBE para completar los datos de estimación

Instrucciones: Durante la fase de planificación, complete las columnas "Estimadas" en el formulario anterior, luego use el Asistente PROBE para completar los campos a continuación.

Tamaño agregado (A):	$A = BA + PA$	<b>TAMAÑO</b>	<b>HORA</b>
Tamaño de proxy estimado (E):	$E = BA + PA + M$	285	
Base estimada de PROBE utilizada: (A, B, C o D)		335	
Correlación: ( $r^2$ )		D	D
Parámetros de regresión:	$B_0$ (tamaño y tiempo)	N/A	0
Parámetros de regresión:	$B_1$ (tamaño y tiempo)	N/A	0
Tamaño agregado y modificado proyectado (P):	$P = B_0 + B_1 \text{ tamaño} * E$	335	
Tamaño total estimado (T):	$T = P + B - D - M + R$	1252	
Nuevo total reutilizable estimado (NR):	(suma de elementos NR)	0	
Tiempo de desarrollo total estimado:	$\text{Tiempo} = B_0 \text{ tiempo} + B_1 \text{ vez} * E$	?????	
Rango de predicción:	Distancia	N/A	0.00
Intervalo de predicción superior:	$UPI = \text{Rango } P +$	N/A	0.00
Intervalo de predicción más bajo:	$LPI = P - \text{Rango}$	N/A	0.00
Porcentaje de Intervalo de Predicción:		N/A	0%

[Ver el informe PROBE](#)



### Ayudante PROBE /BOLSA DE TRABAJO/PSP1

La estimación de tamaño y tiempo son habilidades complejas, mejoradas a lo largo de experiencias. Como trataras de mejorar tus estimaciones, tu primera meta debe ser una estimación precisa.

[Continuar](#)

### PROBE - /BOLSA DE TRABAJO/PSP1

#### Paso 1 - Verificar Tamaño Estimado de Proxy

De tu Plantilla de Estimación de Tamaño, tu Tamaño Estimado de Proxy es 335.

Si se desea modificar tu Tamaño Estimado de Proxy, debes regresar a la Plantilla de Estimación de Tamaño ahora y hacer los cambios necesarios.

Si estas satisfecho con tu Tamaño Estimado de Proxy, oprime el botón Continuar.

[Atrás](#) [Continuar](#)

### PROBE - /BOLSA DE TRABAJO/PSP1

#### Paso 2 - Verificador de Datos Históricos

No se tienen datos históricos.

Si la información de arriba es correcta, oprimir el botón continuar.

[Atrás](#) [Continuar](#)



## PROBE - /BOLSA DE TRABAJO/PSP1

### Paso 3 - Tamaño

Para crear tu estimado final tamaño, utiliza tu criterio ingenieril para elegir uno de los métodos PROBE:

• 335 LOC

Tu mejor opción para estimar tamaño puede ser Método PROBE D. Utiliza tu criterio ingenieril para estimar LOC Nuevas & Cambiadas requerido para este proyecto e introdúcelo en el campo de la izquierda.

No puedes usar Método PROBE A. No se tienen suficientes datos históricos.

No puedes usar Método PROBE B. No se tienen suficientes datos históricos.

No puedes usar Método PROBE C1. No se tienen suficientes datos históricos.

No puedes usar Método PROBE C2. No se tienen suficientes datos históricos.

Elige tu estimado tamaño de las opciones de arriba, y da clic en el botón Continuar.

Atrás Continuar

## PROBE - /BOLSA DE TRABAJO/PSP1

### Paso 4 - Tiempo

Para crear tu estimado final tiempo, utiliza tu criterio ingenieril para elegir uno de los métodos PROBE:

• 4 Horas

Tu mejor opción para estimar tiempo puede ser Método PROBE D. Utiliza tu criterio ingenieril para estimar tiempo de desarrollo total requerido para este proyecto e introdúcelo en el campo de la izquierda.

No puedes usar Método PROBE A. No se tienen suficientes datos históricos.

No puedes usar Método PROBE B. No se tienen suficientes datos históricos.

No puedes usar Método PROBE C1. No se tienen suficientes datos históricos.

No puedes usar Método PROBE C2. No se tienen suficientes datos históricos.

No puedes usar Método PROBE C3. No se tienen suficientes datos históricos.

Elige tu estimado tiempo de las opciones de arriba, y da clic en el botón Continuar.

Atrás Continuar

## PROBE - /BOLSA DE TRABAJO/PSP1

### Paso 5 - Chequeo de Estimados

Has estimado que este proyecto requerirá:

- 335 LOC Nuevas & Cambiadas
- 4 Horas Totales

Estimas obtener para tiempo y tamaño una productividad planeada de 83.8 LOC Hr.

**Felicitaciones!** Has completado el proceso PROBE. Oprimir el botón Finalizar para cerrar esta ventana.

Atrás Finalizar

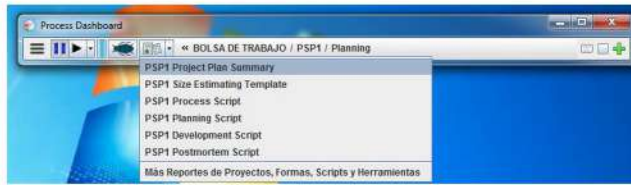
Ya tenemos la estimación de tamaño con el método PROBE

Instrucciones: Durante la fase de planificación, complete las columnas "Estimadas" en el formulario anterior, luego use el [Asistente PROBE](#) para completar los campos a continuación.

		TAMAÑO	HORA
Tamaño agregado (A):	$A = BA + PA$	285	
Tamaño de proxy estimado (E):	$E = BA + PA + M$	335	
Base estimada de PROBE utilizada: (A, B, C o D)		D	
Correlación: ( $r^2$ )		N/A	N/A
Parámetros de regresión:	$B_0$ (tamaño y tiempo)	N/A	N/A
Parámetros de regresión:	$B_1$ (tamaño y tiempo)	N/A	N/A
Tamaño agregado y modificado proyectado (P):	$P = B_0 + B_1 \text{ tamaño} * E$	335	
Tamaño total estimado (T):	$T = P + B - D - M + R$	1252	
Nuevo total reutilizable estimado (NR):	(suma de elementos NR)	D	
Tiempo de desarrollo total estimado:	$\text{Tiempo} = B_0 + B_1 \text{ vez} * E$		4:00
Rango de predicción:	Distancia	N/A	N/A
Intervalo de predicción superior:	$LPI = \text{Rango} P +$	N/A	N/A
Intervalo de predicción más bajo:	$LPI = P - \text{Rango}$	N/A	N/A
Porcentaje de Intervalo de Predicción:		N/A	N/A

Ver el [informe PROBE](#)

Accedemos al PROYECT PLAN SUMMARY en la herramienta “Process Dashboard”



Desde el PROYECT PLAN SUMMARY se verifica la información cargada anteriormente

**PSP1 Project Plan Summary**

[Project Settings](#)  
[Overall Metrics](#)  
[Time in Phase](#)  
[Defects Injected](#)  
[Defects Removed](#)

[Versión sin marcos](#)  
(Mejor para imprimir)

---

**/BOLSA DE TRABAJO/PSP1**

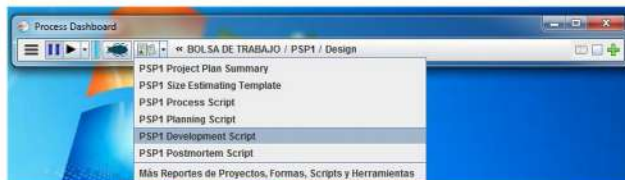
**PSP1 Project Plan Summary**

**Overall Metrics**

Summary	Plan	Actual	A la Fecha
<i>Size Hour</i>	83.8	#VALUE!	25
<b>Program Size</b>			
Base (B)	1000	0	0
Deleted (D)	100	0	0
Modified (M)	50	0	0
Added (A)	286	#VALUE!	0.0
Reused (R)	67	0	0.0
Added and Modified (A+M)	335	#VALUE!	0.0
Total Size (T)	1252	?????	0.0
Total New Reusable	0	0	0
<i>Estimated Proxy Size (E)</i>	335	0	0

## Desarrollo

Se selecciona el SCRIPT DE DISEÑO de la herramienta “Process Dashboard”



## Script de desarrollo PSP1

<b>Propósito</b>	Para guiar el desarrollo de pequeños programas	
<b>Criterio para entrar</b>	<ul style="list-style-type: none"> <li>• <a href="#">Declaración de requisitos</a></li> <li>• Formulario de <a href="#">resumen del plan del proyecto</a> con el tamaño estimado del programa y el tiempo de desarrollo</li> <li>• Registros de registro de <a href="#">tiempo y defectos</a></li> <li>• <a href="#">Estándar de tipo de defecto y estándar de codificación</a></li> </ul>	
<b>Paso</b>	<b>Ocupaciones</b>	<b>Descripción</b>
1	Diseño	<ul style="list-style-type: none"> <li>• Revise los requisitos y produzca un diseño para cumplirlos.</li> <li>• Registre en el <a href="#">registro de registro de defectos</a> cualquier defecto encontrado en los requisitos.</li> <li>• Tiempo de registro en el <a href="#">registro de tiempo de grabación</a></li> </ul>
2	Código	<ul style="list-style-type: none"> <li>• Implementar el diseño siguiendo el <a href="#">estándar de codificación</a></li> <li>• Registre en el <a href="#">registro de grabación de defectos</a> cualquier requisito o defecto de diseño encontrado.</li> <li>• Tiempo de registro en el <a href="#">registro de tiempo de grabación</a></li> </ul>
3	Compilar	<ul style="list-style-type: none"> <li>• Compile el programa hasta que no haya errores de compilación.</li> <li>• Repare todos los defectos encontrados</li> <li>• Grabar defectos en el <a href="#">registro de registro de defectos</a></li> <li>• Tiempo de registro en el <a href="#">registro de tiempo de grabación</a></li> </ul>
4	Prueba	<ul style="list-style-type: none"> <li>• Pruebe hasta que todas las pruebas se ejecuten sin error.</li> <li>• Repare todos los defectos encontrados</li> <li>• Grabar defectos en el <a href="#">registro de registro de defectos</a></li> <li>• Tiempo de registro en el <a href="#">registro de tiempo de grabación</a></li> <li>• <b>Complete una plantilla de Informe de prueba en las pruebas realizadas y los resultados obtenidos.</b></li> </ul>
<b>Criterio de salida</b>	<ul style="list-style-type: none"> <li>• Un programa completamente probado que cumple con el <a href="#">estándar de codificación</a></li> <li>• <b>Completado la plantilla Informe de prueba</b></li> <li>• Registros de registro de <a href="#">tiempo y defectos</a> completados</li> </ul>	

[Siguiente: Postmortem](#)

[Parte superior](#)

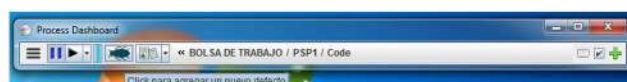
### Diseño

Se registra los tiempos del diseño en la herramienta “Process Dashboard”



### Código

Se ingresan los errores presentados en la etapa de codificación en la herramienta “Process Dashboard”



Se ingresa los errores presentados en la etapa de codificación estos pueden ser de documentación, interface, base de datos, funciones o procedimientos internos, etc.



## Compilar

Se mide el tiempo de compilación del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO en la herramienta "Process Dashboard"



## Prueba

Se mide el tiempo de pruebas del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO en la herramienta "Process Dashboard"

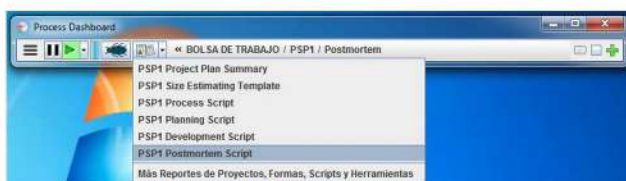


## Post Mortem

Se inicia la etapa de Post Mortem en la herramienta "Process Dashboard"



Elegimos el SCRIPT DE POST MORTEM



## PSP1 Postmortem Script

<b>Propósito</b>	Para guiar el proceso post mortem de PSP
<b>Criterio para entrar</b>	<ul style="list-style-type: none"> <li>Descripción del problema y <a href="#">declaración de requisitos</a></li> <li>Formulario de <a href="#">resumen del plan del proyecto</a> con el tamaño del programa y los datos de tiempo de desarrollo</li> <li><a href="#">Completado la plantilla Informe de prueba</a></li> <li>Registros de registro de <a href="#">tiempo</a> y <a href="#">defectos</a> completados</li> <li>Un programa probado y en ejecución que cumple con los estándares de <a href="#">codificación</a> y conteo de tamaños</li> </ul>
<b>Paso/Ocupaciones</b>	<b>Descripción</b>
1 Grabación de defectos	<ul style="list-style-type: none"> <li>Revise el <a href="#">resumen del plan del proyecto</a> para verificar que se hayan registrado todos los defectos encontrados en cada fase.</li> <li>Usando su mejor recuerdo, registre cualquier defecto omitido.</li> </ul>
2 Consistencia de datos de defecto	<ul style="list-style-type: none"> <li>Verifique que los datos de cada defecto en el <a href="#">registro de registro de defectos</a> sean precisos y completos.</li> <li>Verifique que el número de defectos inyectados y eliminados por fase sea razonable y correcto.</li> <li>Con su mejor recuerdo, corrija los datos de defectos faltantes o incorrectos.</li> </ul>
3 tamaño	<ul style="list-style-type: none"> <li>Cuenta el tamaño del programa completo.</li> <li><i>Determine el tamaño de la base, borrado, modificado, adiciones base, reutilizado, nuevo código reutilizable y partes agregadas. (Si aún no cuenta con una herramienta para medir LOC eliminados, modificados y agregados, puede encontrar útil <a href="#">esta herramienta</a>).</i></li> <li><i>Ingrese estos datos en la <a href="#">plantilla de Estimación del tamaño</a>.</i></li> <li><i>Determine el tamaño total del programa</i></li> <li><i>Ingrese esta información en el formulario <a href="#">Resumen del plan del proyecto</a>.</i></li> </ul>
4 Hora	<ul style="list-style-type: none"> <li>Revise el <a href="#">registro de registro de tiempo</a> completo para ver si hay errores u omisiones.</li> <li>Usando su mejor recuerdo, corrija cualquier dato de tiempo faltante o incompleto.</li> </ul>
<b>Criterio de salida</b>	<ul style="list-style-type: none"> <li>Un programa completamente probado que cumple con los estándares de <a href="#">codificación</a> y conteo de tamaño</li> <li><a href="#">Completado la plantilla Informe de prueba</a></li> <li>Formulario de <a href="#">resumen del plan de proyecto</a> completado</li> <li><a href="#">Formularios PIP</a> completos que describen los problemas del proceso, sugerencias de mejora y lecciones aprendidas</li> <li>Registros de registro de <a href="#">tiempo</a> y <a href="#">defectos</a> completados</li> </ul>

[Parte superior](#)

## Accedemos al RESUMEN DEL PLAN DEL PROYECTO

**PSP1**

**Project Plan Summary**

[Project Settings](#)

[Overall Metrics](#)

[Time In Phase](#)

[Defects Injected](#)

[Defects Removed](#)

---

[Versión sin marcos](#)  
(Mejor para imprimir)

**/BOLSA DE TRABAJO/PSP1**

**PSP1 Project Plan Summary**

✖ Debe ingresar datos actuales de tamaño para este proyecto.

**Overall Metrics**

Summary	Plan	Actual	A la Fecha
Size/Hour	03.0	#VALUE!	25
<b>Program Size</b>			
Base (B)	1000	0	0
Deleted (D)	100	0	0
Modified (M)	50	0	0
Added (A)	285	#VALUE!	0.0
Reused (R)	67	0	0
Added and Modified (A+M)	335	#VALUE!	0.0
Total Size (T)	1252	?????	0.0
Total New Reusable	0	0	0
Estimated Proxy Size (E)	335	0	0

## Observamos los DEFECTOS INYECTADOS hasta la fecha

[Project Settings](#)

[Overall Metrics](#)

[Time In Phase](#)

[Defects Injected](#)

[Defects Removed](#)

---

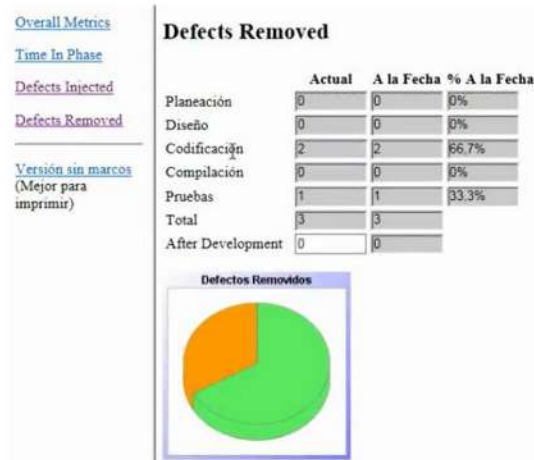
[Versión sin marcos](#)  
(Mejor para imprimir)

**Defects Injected**

	Actual	A la Fecha	% A la Fecha
Antes de Desarrollo	0	0	0%
Planeación	0	0	0%
Diseño	2	2	66.7%
Codificación	1	1	33.3%
Compilación	0	0	0%
Pruebas	0	0	0%
Total	3	3	

**Defectos Inyectados**

Observamos los DEFECTOS REMOVIDOS hasta la fecha



Accedemos a la PLANTILLA DE ESTIMACIÓN DE TAMAÑO para continuar necesitamos métricas del código del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO.

Para conocer las métricas de código utilizamos la herramienta CLOC que sirve para medir las líneas de código.

```
Cmder
D:\TEMP\cloc-1.74.exe .
1464 text files.
881 unique files.
1267 files ignored.

github.com/AlDanial/cloc v 1.74 T=11.00 s (57.9 files/s, 26926.2 lines/s)
-----
Language files blank comment code
-----
C# 278 20812 6246 164269
XML 33 51 1 65014
XSD 3 0 0 15209
SQL 284 984 165 8925
MSBuild script 67 0 443 7223
ASP.NET 10 970 1 1900
HTML 9 198 66 1181
CSS 2 126 46 1050
DOS Batch 25 50 29 278
XAML 3 3 0 55
Visual Basic 1 13 6 41
Cucumber 2 0 0 13
-----
SUM: 637 23127 7003 260058
-----
```

Rellenamos los datos obtenidos anterior mente sobre las métricas del código

## / BOLSA DE TRABAJO / PSP1

### Plantilla de estimación de tamaño [\(instrucciones ...\)](#)

Dueño del proyecto FERNANDO SILVERIO BRAVO

Medida de tamaño LOC

PIEZAS BASE	Estimado				Real			
	BASE	ELIMINADO	MODIFICADO	ADICIONAL	BASE	ELIMINADO	MODIFICADO	ADICIONAL
Modulo Alumno	1000	100	50	200	1005	98	48	202
<b>agregue más filas para partes de base ... Total:</b>	<b>1000</b>	<b>100</b>	<b>50</b>	<b>200</b>	<b>1005</b>	<b>98</b>	<b>48</b>	<b>202</b>

ADICIONES DE PIEZAS	ID	ARTICULO	Estimado		Real	
			TAMAJO	SR	TAMAJO	SR
Clase Desviación Estandar	1	Medio	11.3		12	
Leer Ficheros	2	Pequeña	24.1		25	
Calculo Tamaños Relativos	2	Grande	49.3		48	
<b>agregue más filas para agregar partes ... Total:</b>			<b>84.7</b>		<b>85</b>	

PIEZAS REUTILIZADAS	Estimado TAMAJO	Real TAMAJO
Clase IntervaloProduccion	67	16
<b>agregue más filas para piezas reutilizadas ... Total:</b>	<b>67</b>	<b>16</b>

TAMAÑO TOTAL	Real TAMAJO
<b>Tamaño real del producto terminado:</b>	<b>1350</b>

Accedemos a la PLANTILLA INFORME DE PRUEBA que será generada por la herramienta “Process Dashboard”

### Introducir Información del Archivo

Favor de proveer la siguiente información para help locate the 'Test Report'.

PSP Project Document Directory C:\PSP

Please enter the complete path to a directory where you would like to store documents pertaining to this PSP Project.

Test Report Filename reportePsp1.doc

OK

Se rellena el TEST REPORT TEMPLATE generado por la herramienta “Process Dashboard”

Test Report Template	
Name	FERNANDO SILVERIO BRAVO
Date	18-Jul-2016
Project/Task	BOLSA DE TRABAJO PSP1
Test Name	Modulo Alumno
Test Objective	Desarrollar el modulo de administración para los alumnos
Test Description	Medir métricas con PSP
Test Conditions	Acorde a los requerimientos y condiciones, las herramientas que se usaran: Lenguajes de programación: • Visual Studio Base de datos: • Sql Server
Expected Results	Mejorar calidad de codificación
Actual Results	Se redujo en 5% las líneas de código con referencia al modulo inicial

## Completamos el Resumen Del Plan De Proyecto

**PSP1**  **/BOLSA DE TRABAJO/PSP1**  
**Project Plan Summary**  
[Project Settings](#)  
[Overall Metrics](#)  
[Time In Phase](#)  
[Defects Injected](#)  
[Defects Removed](#)  
[Versión sin marcos \(Mejor para imprimir\)](#)

**PSP1 Project Plan Summary**

**Overall Metrics**



Summary	Plan	Actual	A la Fecha
<i>Size/Hour</i>	83.8	2192	25
<b>Program Size</b>			
Base (B)	1000	1005	0
Deleted (D)	100	98	0
Modified (M)	50	48	0
Added (A)	285	427	0.0
Reused (R)	67	16	0
Added and Modified (A+M)	335	475	0.0
Total Size (T)	1252	1350	0.0
Total New Reusable	0	0	0
<i>Estimated Proxy Size (E)</i>	335	335	0

## Acedemos a los TIEMPOS DE FASE

[Project Settings](#)  
[Overall Metrics](#)  
[Time In Phase](#)  
[Defects Injected](#)  
[Defects Removed](#)  
[Versión sin marcos \(Mejor para imprimir\)](#)

**Time In Phase**

	Plan	Actual	A la Fecha	% A la Fecha
Planeación	0.21	0.25	0.10	6.77%
Diseño	0.53	0.36	0.25	21.9%
Codificación	1.38	1.12	0.47	41.2%
Complación	0.00	0.00	0.00	0%
Pruebas	0.40	0.19	0.19	16.7%
Postmortem	0.27	0.18	0.13	11.4%
Total	4.00	2.50	1.54	

Planned Tiempo:  Actual Tiempo: 

## Acedemos al RESUMEN DEL PLAN DEL PROYECTO

**PSP1**  **/BOLSA DE TRABAJO/PSP1**  
**Project Plan Summary**  
[Project Settings](#)  
[Overall Metrics](#)  
[Time In Phase](#)  
[Defects Injected](#)  
[Defects Removed](#)  
[Versión sin marcos \(Mejor para imprimir\)](#)

**PSP1 Project Plan Summary**

**Overall Metrics**

Summary	Plan	Actual	A la Fecha
<i>Size/Hour</i>	83.8	2192	25
<b>Program Size</b>			
Base (B)	1000	1005	0
Deleted (D)	100	98	0
Modified (M)	50	48	0
Added (A)	285	427	0.0
Reused (R)	67	16	0
Added and Modified (A+M)	335	475	0.0
Total Size (T)	1252	1350	0.0
Total New Reusable	0	0	0
<i>Estimated Proxy Size (E)</i>	335	335	0

Acedemos al FORMULARIOS PIP que será generada por la herramienta “Process Dashboard”

### Introducir Información del Archivo

Favor de proveer la siguiente información para create the 'PIP Form'.

PSP Project Document Directory:

Please enter the complete path to a directory where you would like to store documents pertaining to this PSP Project.

PIP Form Filename:



Se rellena el PROCESS IMPROVEMENT PROPOSAL (PIP) generado por la herramienta "Process Dashboard"

**Process Improvement Proposal (PIP)**

Name	FERNANDO SILVERIO BRAVO	Date	18-jul-2018
Project/Task	/BOLSA DE TRABAJO/PSP1		
Process	PSP1	Elements	

**PIP # Problem Description**

1. La interface no muestra los datos integros de la consulta

**PIP # Proposal Description**

1. Modificar el css de interface general

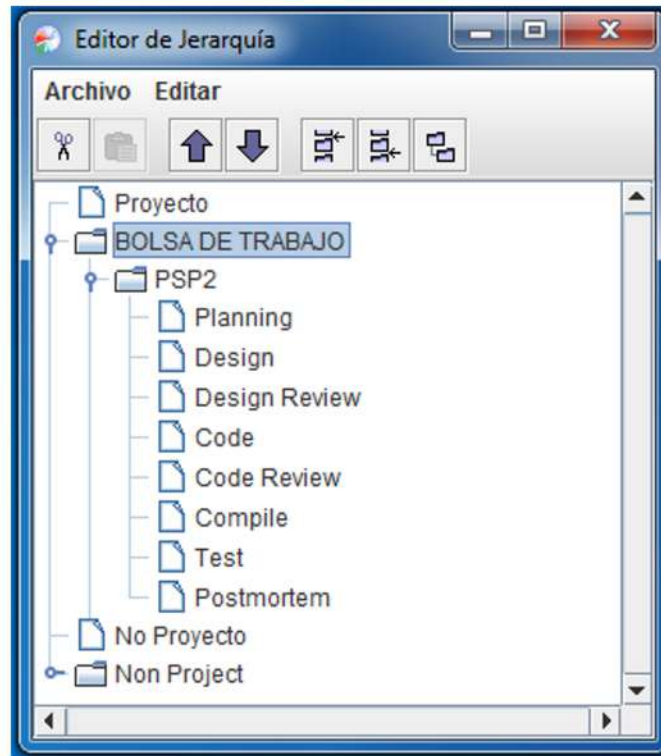
**Notes and Comments**

|

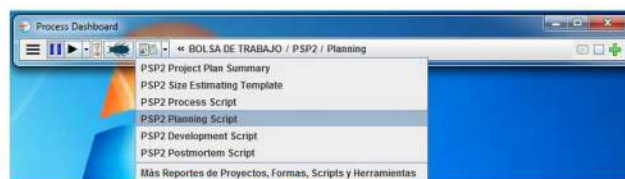
Estas métricas y documentación obtenidas en PSP 1 se utilizan para mejorar la calidad del proceso en PSP 1.1 a través de planificación de tareas y planeación de tiempo orientados a mejorar la distribución de trabajo en el tiempo estimado.

## INICIAR EL PSP2

Para iniciar con el nivel PSP2: Proceso de administración de calidad personal manejar defectos de programación la herramienta “Process Dashboard” nos permite gestionar las distintas categorías requeridas para completar dicho nivel.



Se elige la herramienta que muestra el SCRIPT DE PLANIFICACIÓN de PSP



## Se muestra el SCRIPT DE PLANIFICACIÓN de PSP2

### Script de planificación PSP2

<b>Propósito</b>	Para guiar el proceso de planificación de PSP
<b>Criterio para entrar</b>	<ul style="list-style-type: none"> <li>Descripción del problema</li> <li>Formulario de <a href="#">resumen del plan de proyecto PSP2</a></li> <li>Plantillas de <a href="#">estimación de tamaño, planificación de tareas y programación</a></li> <li><a href="#">Datos históricos de tamaño y tiempo (estimados y reales)</a></li> <li><a href="#">Registro de tiempo de grabación</a></li> </ul>
<b>Paso/Ocupaciones</b>	<b>Descripción</b>
1 Requisitos del programa	<ul style="list-style-type: none"> <li>Producir u obtener una <a href="#">declaración de requisitos</a> para el programa.</li> <li>Asegúrese de que la declaración de requisitos sea clara y no ambigua.</li> <li>Resuelva cualquier pregunta.</li> </ul>
2 Estimación del tamaño	<ul style="list-style-type: none"> <li>Producir un programa de <a href="#">diseño conceptual</a></li> <li>Use el <a href="#">método PROBE</a> para estimar el tamaño agregado y modificado de este programa.</li> <li>Complete la <a href="#">plantilla de Estimación del tamaño</a> y el formulario <a href="#">Resumen del plan del proyecto</a>.</li> </ul>
3 Estimación de recursos	<ul style="list-style-type: none"> <li>Use el <a href="#">método PROBE</a> para estimar el tiempo requerido para desarrollar este programa.</li> <li>Usando el% hasta la fecha del último programa desarrollado como guía, <a href="#">el tablero distribuirá automáticamente</a> el tiempo de desarrollo sobre las fases planificadas del proyecto.</li> </ul>
4 Planificación de tareas y horarios	<ul style="list-style-type: none"> <li>Para proyectos que duran varios días o más, complete las <a href="#">plantillas Planificación de tareas y Planificación de programación</a>.</li> </ul>
5 Estimación de defectos	<ul style="list-style-type: none"> <li><i>En función de los datos actualizados sobre defectos por unidad de tamaño agregado y modificado, <a href="#">el tablero automáticamente calculará los defectos totales que se encontrarán en este programa.</a></i></li> <li><i>Según los datos de su fecha hasta el momento, <a href="#">el tablero automáticamente estimará el número de defectos que se inyectarán y eliminarán por fase.</a></i></li> </ul>
<b>Criterio de salida</b>	<ul style="list-style-type: none"> <li><a href="#">Declaración de requisitos</a> documentados</li> <li><a href="#">Diseño conceptual</a> del programa</li> <li>Completado <a href="#">plantilla de la estimación del tamaño</a></li> <li>Para proyectos que duran varios días o más, <a href="#">plantillas de tareas y planificación planificadas</a></li> <li>Formulario de <a href="#">resumen del plan del proyecto</a> completo con el tamaño estimado del programa, el tiempo de desarrollo y <i>los datos de defectos</i></li> <li><a href="#">Registro de registro de tiempo</a> completado</li> </ul>

Accedemos a DATOS HISTÓRICOS DE TAMAÑO Y TIEMPO (ESTIMADOS Y REALES) para verificar los datos guardados hasta la fecha

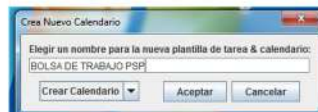
#### Datos PSP Históricos

Tiempo y Tamaño Actuales y Estimados

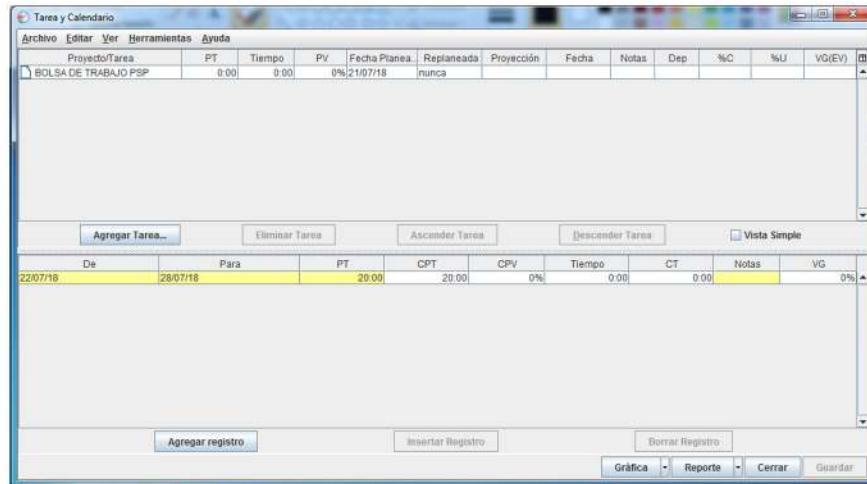
Proyecto/Tarea	Est LOC de Obj	Est LOC N&C	LOC Act N&C	Min Est	Min Act
PSP			0	180	114
PSP	153	153	173	240	172

[Exportar a Excel](#)

Accedemos a PLANTILLAS PLANIFICACIÓN DE TAREAS Y PLANIFICACIÓN DE PROGRAMACIÓN donde generamos un nuevo calendario.



En función de los datos actualizados sobre defectos por unidad de tamaño agregado y modificado, el tablero automáticamente calculará los defectos totales que se encontrarán en este programa.



## Completamos la PLANTILLA DE ESTIMACIÓN DE TAMAÑO

### / BOLSA DE TRABAJO / PSP2

#### Plantilla de estimación de tamaño [\(instrucciones ...\)](#)

Dueño del proyecto: FERNANDO SILVERIO BRAVO

Medida de tamaño: LOC

PIEZAS BASE	Estimado				Real			
	BASE	ELIMINADO	MODIFICADO	ADICIONAL	BASE	ELIMINADO	MODIFICADO	ADICIONAL
Modulo Empresa	1500	350	250	3200	0	0	0	0
<a href="#">agregue más filas para partes de base ...</a>	<b>Total:</b> 1500 350 250 3200				0 0 0 0			

ADICIONES DE PIEZAS	ID	ARTICULOS	DEL TAMAÑO	TAMAÑO	VR	Real	
						TAMAÑO	ARTICULOS VR
Registro de Propuestas	Datos		Grande	16.3			
<a href="#">agregue más filas para agregar partes ...</a>	<b>Total:</b> 16.3					0	

PIEZAS REUTILIZADAS		Estimado	Real
		TAMAÑO	TAMAÑO
Leer Archivos		54	0
Calculos Desviacion Estandar		46	
<a href="#">agregue más filas para piezas reutilizadas ...</a>			
<b>Total:</b>		100	0

TAMAÑO TOTAL	Real
Tamaño real del producto terminado:	TAMAÑO

Utilizamos el ASISTENTE PROBE para completar los datos de estimación

Instrucciones: Durante la fase de planificación, complete las columnas "Estimadas" en el formulario anterior, luego use el **Asistente PROBE** para completar los campos a continuación.

Tamaño agregado (A):	$A = BA + PA$	<b>TAMAÑO</b>	<b>HORA</b>
Tamaño de proxy estimado (E):	$E = BA + PA + M$	3236	
Base estimada de PROBE utilizada: (A, B, C o D)		0	0
Correlación: ( $r^2$ )		0	0
Parámetros de regresión:	$B_0$ (tamaño y tiempo)	0	0
Parámetros de regresión:	$B_1$ (tamaño y tiempo)	0	0
Tamaño agregado y modificado proyectado (P):	$P = B_0 + B_1 \text{ tamaño} * E$	?????	
Tamaño total estimado (T):	$T = P + B - D - M + R$	#VALUE!	
Nuevo total reutilizable estimado (NR):	(suma de elementos NR)	0	
Tiempo de desarrollo total estimado:	$\text{Tiempo} = B_0 \text{ tiempo} + B_1 \text{ vez} * E$	?????	
Rango de predicción:	Distancia	0	0.00
Intervalo de predicción superior:	$UPI = \text{Rango } P +$	0	0.00
Intervalo de predicción más bajo:	$LPI = P - \text{Rango}$	0	0.00
Porcentaje de Intervalo de Predicción:		0%	0%

[Ver el informe PROBE](#)



## Ayudante PROBE /BOLSA DE TRABAJO/PSP2

La estimación de tamaño y tiempo son habilidades complejas, mejoradas a lo largo de experiencias. Como trataras de mejorar tus estimaciones, tu primera meta debe ser una estimación precisa.

[Continuar](#)

## PROBE - /BOLSA DE TRABAJO/PSP2

### Paso 1 - Verificar Tamaño Estimado de Proxy

De tu Plantilla de Estimación de Tamaño, tu Tamaño Estimado de Proxy es 3466.

Si se desea modificar tu Tamaño Estimado de Proxy, debes regresar a la Plantilla de Estimación de Tamaño ahora y hacer los cambios necesarios.

Si estas satisfecho con tu Tamaño Estimado de Proxy, oprime el botón Continuar.

[Atrás](#) [Continuar](#)

## PROBE - /BOLSA DE TRABAJO/PSP2

### Paso 2 - Verificador de Datos Históricos

No se tienen datos históricos.

Si la información de arriba es correcta, oprimir el botón continuar.

[Atrás](#) [Continuar](#)

## PROBE - /BOLSA DE TRABAJO/PSP2

### Paso 3 - Tamaño

Para crear tu estimado final tamaño, utiliza tu criterio ingenieril para elegir uno de los métodos PROBE:

3466 LOC

Tu mejor opción para estimar tamaño puede ser Método PROBE D. Utiliza tu criterio ingenieril para estimar LOC Nuevas & Cambiadas requerido para este proyecto e introducelo en el campo de la izquierda.

No puedes usar Método PROBE A. No se tienen suficientes datos históricos.

No puedes usar Método PROBE B. No se tienen suficientes datos históricos.

No puedes usar Método PROBE C1. No se tienen suficientes datos históricos.

No puedes usar Método PROBE C2. No se tienen suficientes datos históricos.

Elige tu estimado tamaño de las opciones de arriba, y da clic en el botón Continuar.

[Atrás](#) [Continuar](#)

## PROBE - /BOLSA DE TRABAJO/PSP2

### Paso 4 - Tiempo

Para crear tu estimado final tiempo, utiliza tu criterio ingenieril para elegir uno de los métodos PROBE:

6 Horas

Tu mejor opción para estimar tiempo puede ser Método PROBE D. Utiliza tu criterio ingenieril para estimar tiempo de desarrollo total requerido para este proyecto e introducelo en el campo de la izquierda.

No puedes usar Método PROBE A. No se tienen suficientes datos históricos.

No puedes usar Método PROBE B. No se tienen suficientes datos históricos.

No puedes usar Método PROBE C1. No se tienen suficientes datos históricos.

No puedes usar Método PROBE C2. No se tienen suficientes datos históricos.

No puedes usar Método PROBE C3. No se tienen suficientes datos históricos.

Elige tu estimado tiempo de las opciones de arriba, y da clic en el botón Continuar.

[Atrás](#) [Continuar](#)

## PROBE - /BOLSA DE TRABAJO/PSP2

### Paso 5 - Chequeo de Estimados

Has estimado que este proyecto requerirá:

- 3466 LOC Nuevas & Cambiadas
- 6 Horas Totales

Estimas obtener para tiempo y tamaño una productividad planeada de 578 LOC Hr.

**Felicitaciones!** Has completado el proceso PROBE. Oprimir el botón Finalizar para cerrar esta ventana.

[Atrás](#) [Finalizar](#)

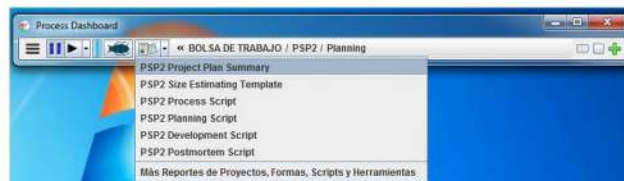
## Ya tenemos la estimación de tamaño con el método PROBE

Instrucciones: Durante la fase de planificación, complete las columnas "Estimadas" en el formulario anterior, luego use el [Asistente PROBE](#) para completar los campos a continuación.

Tamaño agregado (A):	$A = BA + PA$	<b>TAMAÑO</b>	<b>HORA</b>
Tamaño de proxy estimado (E):	$E = BA + PA + M$	3216	
Base estimada de PROBE utilizada: (A, B, C o D)		3466	
Correlación: ( $r^2$ )		D	D
Parámetros de regresión:	$B_0$ (tamaño y tiempo)	N/A	N/A
Parámetros de regresión:	$B_1$ (tamaño y tiempo)	N/A	N/A
Tamaño agregado y modificado proyectado (P):	$P = B_0 \text{ tamaño} + B_1 \text{ tamaño} * E$	3466	
Tamaño total estimado (T):	$T = P + B - D - M + R$	4466	
Nuevo total reutilizable estimado (NR):	(suma de elementos NR)	0	
Tiempo de desarrollo total estimado:	$\text{Tiempo} = B_0 \text{ tiempo} + B_1 \text{ vez} * E$		6:00
Rango de predicción:	Distancia	N/A	N/A
Intervalo de predicción superior:	$UPI = \text{Rango } P +$	N/A	N/A
Intervalo de predicción más bajo:	$LPI = P - \text{Rango}$	N/A	N/A
Porcentaje de Intervalo de Predicción:		N/A	N/A

[Ver el informe PROBE](#)

Accedemos al PROYECT PLAN SUMMARY en la herramienta "Process Dashboard"



Desde el PROYECT PLAN SUMMARY se verifica la información cargada anteriormente

<h3>Project Plan Summary</h3> <ul style="list-style-type: none"> <li><a href="#">Project Settings</a></li> <li><a href="#">Overall Metrics</a></li> <li><a href="#">Program Size</a></li> <li><a href="#">Time In Phase</a></li> <li><a href="#">Defects Injected</a></li> <li><a href="#">Defects Removed</a></li> <li><a href="#">Defect Removal Efficiency</a></li> <li><a href="#">Versión sin marcos</a></li> </ul>	<h3>PSP2 Project Plan Summary</h3> <h4>Program Size</h4> <table border="1"> <thead> <tr> <th></th> <th>Plan</th> <th>Actual</th> <th>A la Fecha</th> </tr> </thead> <tbody> <tr> <td>Base (B)</td> <td>150</td> <td>0</td> <td>100</td> </tr> <tr> <td>Deleted (D)</td> <td>35</td> <td>0</td> <td>26</td> </tr> <tr> <td>Modified (M)</td> <td>25</td> <td>0</td> <td>23</td> </tr> <tr> <td>Added (A)</td> <td>95.2</td> <td>#VALUE!</td> <td>150</td> </tr> <tr> <td>Reused (R)</td> <td>100</td> <td>0</td> <td>67</td> </tr> <tr> <td>Added and Modified (A+M)</td> <td>120</td> <td>#VALUE!</td> <td>173</td> </tr> <tr> <td>Total Size (T)</td> <td>310</td> <td>?????</td> <td>291</td> </tr> <tr> <td>Total New Reusable</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>Estimated Proxy Size (E)</td> <td>106</td> <td>0</td> <td>173</td> </tr> </tbody> </table>		Plan	Actual	A la Fecha	Base (B)	150	0	100	Deleted (D)	35	0	26	Modified (M)	25	0	23	Added (A)	95.2	#VALUE!	150	Reused (R)	100	0	67	Added and Modified (A+M)	120	#VALUE!	173	Total Size (T)	310	?????	291	Total New Reusable	0	0	0	Estimated Proxy Size (E)	106	0	173
	Plan	Actual	A la Fecha																																						
Base (B)	150	0	100																																						
Deleted (D)	35	0	26																																						
Modified (M)	25	0	23																																						
Added (A)	95.2	#VALUE!	150																																						
Reused (R)	100	0	67																																						
Added and Modified (A+M)	120	#VALUE!	173																																						
Total Size (T)	310	?????	291																																						
Total New Reusable	0	0	0																																						
Estimated Proxy Size (E)	106	0	173																																						

## Accedemos a PROGRAM SIZE

PSP2 Project Plan Summary			
Program Size			
	Plan	Actual	A la Fecha
Base (B)	1500	0	0
Deleted (D)	350	0	0
Modified (M)	250	0	0
Added (A)	3216	#VALUE!	0,0
Reused (R)	100	0	0
Added and Modified (A-M)	3466	#VALUE!	0,0
Total Size (T)	4466	?????	0,0
Total New Reusable	0	0	0
Estimated Proxy Size (E)	3466	0	0

## Accedemos a TIME IN PHASE

PSP2 Project Plan Summary				
Time In Phase				
	Plan	Actual	A la Fecha	% A la Fecha
Planeación	0:15	0:10	0:35	12,2%
Diseño	0:25	0:00	1:01	21,3%
<i>Revisión de Diseño</i>	0:13	0:00	0:00	0%
Codificación	0:50	0:00	1:59	41,6%
<i>Revisión de Codificación</i>	0:30	0:00	0:00	0%
Compilación	0:00	0:00	0:00	0%
Pruebas	0:10	0:00	0:38	13,3%
Postmortem	0:14	0:00	0:33	11,5%
Total	2:00	0:10	4:46	

Planned Tiempo	Actual Tiempo

## Accedemos a DEFECTS INJECTED

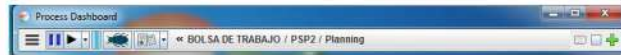
PSP2 Project Plan Summary				
Defects Injected				
	Plan	Actual	A la Fecha	% A la Fecha
Antes de Desarrollo	0	0	0	0%
Planeación	0	0	0	0%
Diseño	89,3	0	0,0	30,8%
<i>Revisión de Diseño</i>	0	0	0	0%
Codificación	139	0	0,0	61,5%
<i>Revisión de Codificación</i>	0	0	0	0%
Compilación	17,3	0	0,0	7,69%
Pruebas	0	0	0	0%
Total	225	0	0,0	

## Accedemos a DEFECTS REMOVED

PSP2 Project Plan Summary				
Defects Removed				
	Plan	Actual	A la Fecha	% A la Fecha
Planeación	0	0	0	0%
Diseño	0	0	0	0%
<i>Revisión de Diseño</i>	1,95	0	0	0%
Codificación	0	0	4	66,7%
<i>Revisión de Codificación</i>	1,55	0	0	0%
Compilación	0	0	0	0%
Pruebas	0,67	0	2	33,3%
Total	4,17	0	6	
After Development	0	0	0	

## Planeación

Se mide el tiempo de compilación del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO en la herramienta “Process Dashboard”



## Diseño

Se mide el tiempo de diseño del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO en la herramienta “Process Dashboard”



## Revisiones de diseño

Se mide el tiempo de revisiones diseño del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO en la herramienta “Process Dashboard”



Para la revisión de diseño se tiene en cuenta los ESTÁNDARES DE CODIFICACIÓN (ANEXO 1)



ESTANDAR DE CODIFICACION

<b>Propósito</b>	Este documento tiene como propósito guiar la implementación de los programas que desarrollare utilizando el lenguaje de programación java
<b>Conteo Estándar</b>	<ul style="list-style-type: none"> <li>Cada línea física representa una línea de código.</li> <li>No se cuentan líneas que solo contienen espacios en blanco ni líneas correspondientes a documentación.</li> <li>No se cuentan paquetes.</li> <li>No se tendrá en cuenta líneas autogeneradas por el entorno.</li> </ul>
<b>Campos, variables.</b>	<ul style="list-style-type: none"> <li>Los nombres de campos y variables iniciaran con letra minúscula.</li> <li>Si tienen nombre compuesto este debe ir sin espacios y la palabra que lo compone debe comenzar con caracteres en mayúscula.</li> <li>Los nombres no deben contener caracteres símbolos.</li> <li>No deben empezar por carácter número.</li> <li>Los nombres deben contener como mínimo 3 caracteres y estos deben dar información de lo que contendrá el campo.</li> </ul>
<b>Interfaces, estructuras, clases</b>	<ul style="list-style-type: none"> <li>Debe ajustarse a las reglas de nombres definidos para las variables con la diferencia que los nombres de estos deben iniciar con letra mayúscula.</li> <li>El nombre debe expresar de manera breve la funcionalidad de la clase.</li> <li>Tendrá un encabezado distintivo "SClase", para la clase. Y después de la línea que cierra la clase tendrá una Línea de documentación "///\$DocClass" para identificar el final de la clase.</li> <li>Tendrá un encabezado distintivo "SEntty", para la Estructura.</li> <li>Tendrá un encabezado distintivo "SInter", para la interfaz.</li> </ul>
<b>Bloques de código</b>	Deben tener una sangría de mínimo dos espacios para facilitar la interpretación del código en las líneas que conforman un bloque.
<b>Comentarios</b>	
<b>De Línea</b>	<ul style="list-style-type: none"> <li>Se utilizará una línea por comentario y esta podrá o no finalizar en punto.</li> <li>Se debe prestar atención a la ortografía.</li> <li>Se usaran para describir el funcionamiento de variables, métodos y clases.</li> </ul>
<b>De bloque</b>	<ul style="list-style-type: none"> <li>Inicio con /*</li> <li>En los intermedios iniciara con *</li> <li>Terminará con */</li> <li>Se usará una línea por comentario y esta podrá o no finalizar con punto.</li> <li>Se utilizará para el encabezado el cual contendrá nombre del programa, fecha, nombre de quien lo realiza y una breve descripción.</li> </ul>
<b>Métodos o funciones</b>	<ul style="list-style-type: none"> <li>Los nombres se ajustaran a las reglas definidas para nombrar clases.</li> <li>El nombre debe dar una breve descripción de lo que hace.</li> <li>Tendrá un encabezado distintivo "SMethod", para la clase.</li> <li>Tendrá una línea de comentario en la línea siguiente de la línea de cierre para identificación "///\$DocMethod"</li> </ul>
<b>Sentencias de decisión</b>	<ul style="list-style-type: none"> <li>Utilizar líneas aunque solo contenga una línea.</li> <li>Utilizar una sintaxis explicativa para los campos que están siendo objeto de comparación.</li> </ul>
<b>Sentencias de iteración</b>	<ul style="list-style-type: none"> <li>Documentar las sentencias que generan ruptura.</li> </ul>

Además es necesario rellenar los formatos PSP2 Design Review Checklist, PS2 Code Review Checklist detallados en DOCUMENTOS PSP (Anexo 2)

PSP2 Design Review Checklist

Modulo DOCENTES Fecha 15/09/2016  
 Responsable FERNANDO SILVERIO BRAVO Language ASP.NET

<b>Propósito General</b>	Para guiarlo en la realización de una revisión de diseño efectiva. - Revise el programa completo para cada categoría de lista de verificación; no intente revisar por más de una categoría a la vez! - A medida que completa cada paso de revisión, marque ese elemento en el cuadro de la derecha. - Complete la lista de verificación para un programa o unidad de programa antes de revisar la próxima.				
<b>Completar</b>	Verifique que el diseño cubra todos los requisitos aplicables. - Todos los productos especificados son producidos. - Todas las entradas necesarias están amuebladas. - Se incluyen todos los requisitos necesarios.				XXX
<b>Límites externos</b>	Cuando el diseño asuma o dependa de límites externos, determine si el comportamiento es correcto a valores nominales, límites y más allá de los límites.				X
<b>Lógica</b>	- Verifique que la secuencia del programa sea correcta. Las pilas, listas, etc. están en el orden correcto. La recursión se desarrolla apropiadamente. - Verifique que todos los bucles se inicien, incrementen y finalicen correctamente.				XXX
<b>Límites internos</b>	- Examine cada enunciado condicional y verifique todos los casos. Cuando el diseño asuma o dependa de límites internos, determine si el comportamiento es correcto a valores nominales, límites y más allá de los límites.				X
<b>Casos especiales</b>	Verifica todos los casos especiales. - Asegure el funcionamiento correcto con valores vacíos, completos, mínimos, máximos, negativos y cero para todas las variables. - Proteger contra condiciones fuera de los límites, desbordamiento y subdesbordamiento - Asegurar que las condiciones "imposibles" sean absolutamente imposibles. - Manejar todas las posibles condiciones incorrectas o de error.				XXXX
<b>Uso Funcional</b>	- Verifique que todas las funciones, procedimientos o métodos se entiendan por completo y se usen correctamente. - Verifique que todas las abstracciones con referencia externa estén definidas con precisión.				XX
<b>Consideraciones del sistema</b>	- Verifique que el programa no cause que se excedan los límites del sistema. - Verifique que todos los datos confidenciales provengan de fuentes confiables. - Verifique que todas las condiciones de seguridad cumplan con las especificaciones de seguridad.				XXX
<b>Nombres</b>	Comprueba uso - todos los nombres especiales son claros, definidos y autenticados - los alcances de todas las variables y parámetros son evidentes por sí mismos o definidos - todos los artículos con nombre se usan dentro de sus ámbitos declarados				XXX
<b>Estándares</b>	Asegúrese de que el diseño cumpla con todos los estándares de diseño aplicables.				X

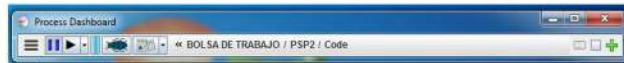
PS2 Code Review Checklist

Modulo DOCENTES Fecha 15/09/2016  
 Responsable FERNANDO SILVERIO BRAVO Language ASP.NET

<b>Propósito General</b>	Para guiarlo en la realización de una revisión de código efectiva. - Revise el programa completo para cada categoría de lista de verificación; no intente revisar por más de una categoría a la vez! - A medida que completa cada paso de revisión, marque ese elemento en el cuadro de la derecha. - Complete la lista de verificación para un programa o unidad de programa antes de revisar el siguiente.				
<b>Completar</b>	Verifique que el código cubre todo el diseño.				X
<b>Incluir</b>	Verifique que las inclusiones estén completas.				X
<b>Inicialización</b>	Verifique la inicialización de variables y parámetros. - al inicio del programa - al inicio de cada ciclo - en la entrada clase / función / procedimiento				XXX
<b>Llamadas</b>	Compruebe los formatos de llamada de función. - punteros - parámetros				XXX
<b>Nombres</b>	- uso de &l Verifique la ortografía y el uso del nombre. - ¿Es consistente? - ¿Está dentro del alcance declarado? - ¿Todas las estructuras y clases usan referencia l?				XXX
<b>Strings</b>	Verifique que todas las cadenas sean - identificadas por punteros - terminadas por NULL				XX
<b>Punteros</b>	Compruebe que todo - los punteros se inicializan NULL - los punteros se eliminan solo después de nuevo - los nuevos punteros siempre se eliminan después del uso				XXX
<b>Formato de salida</b>	Verifica el formato de salida. - El paso de línea es apropiado. - El espaciado es correcto.				XX
<b>() Pairs</b>	Asegúrese de que () sean correctos y coincidan.				X
<b>Operadores Lógicos</b>	- Verifique el uso correcto de ==, !=,  , y así sucesivamente. - Verifica cada función lógica para ()				XX
<b>Comprobación línea por línea</b>	Verifique cada línea de código para - sintaxis de instrucciones - puntuación adecuada				XX
<b>Estándares</b>	Asegúrese de que el código cumpla con los estándares de codificación.				X
<b>Archivo abierto y cerrado</b>	Verifique que todos los archivos sean - debidamente declarado - abrió - cerrado				XXX

## Código

Se mide el tiempo de codificación del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO en la herramienta “Process Dashboard”



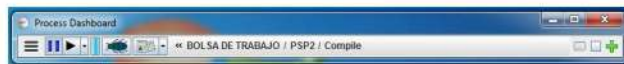
## Revisiones de Código

Se mide el tiempo de revisiones de codificación del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO en la herramienta “Process Dashboard”



## Compilación

Se mide el tiempo de compilación del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO en la herramienta “Process Dashboard”



## Prueba

Se mide el tiempo de prueba del software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO en la herramienta “Process Dashboard”

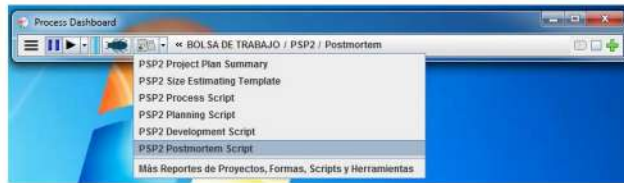


## Post mortem

Se inicia la etapa de Post Mortem en la herramienta “Process Dashboard”



## Elegimos el SCRIPT DE POST MORTEM



### PSP2 Postmortem Script

Propósito	Para guiar el proceso post mortem de PSP	
Criterio para entrar	<ul style="list-style-type: none"> <li>Descripción del problema y <a href="#">declaración de requisitos</a></li> <li>Formulario de <a href="#">resumen del plan del proyecto</a> con tamaño del programa, tiempo de desarrollo y datos de <a href="#">defectos</a></li> <li>Para proyectos que duran varios días o más, <a href="#">plantillas de planificación de tareas</a> y <a href="#">planificación de planificación completa</a></li> <li>Completado la <a href="#">plantilla Informe de prueba</a></li> <li>Completadas <a href="#">revisión de diseño</a> y <a href="#">revisión de código listas de comprobación</a></li> <li>Registros de registro de <a href="#">tiempo</a> y <a href="#">defectos</a> completados</li> <li>Un programa probado y en ejecución que cumple con los estándares de <a href="#">codificación</a> y conteo de tamaños</li> </ul>	
Paso	Ocupaciones	Descripción
1	Grabación de defectos	<ul style="list-style-type: none"> <li>Revise el <a href="#">resumen del plan del proyecto</a> para verificar que se hayan registrado todos los defectos encontrados en cada fase.</li> <li>Usando su mejor recuerdo, registre cualquier defecto omitido.</li> </ul>
2	Consistencia de datos de defecto	<ul style="list-style-type: none"> <li>Verifique que los datos de cada defecto en el <a href="#">registro de registro de defectos</a> sean precisos y completos.</li> <li>Verifique que el número de defectos proyectados y eliminados por fase sea razonable y correcto.</li> <li>Determine el <a href="#">rendimiento del proceso</a> y <a href="#">verifique que el valor sea razonable y correcto</a>.</li> <li>Con su mejor recuerdo, corrija los datos de defectos faltantes o incorrectos.</li> </ul>
3	tamaño	<ul style="list-style-type: none"> <li>Cuente el tamaño del programa completo.</li> <li>Determine el tamaño de la base, borrado, modificado, adiciones base, reutilizado, nuevo código reutilizable y partes agregadas. (Si aún no cuenta con una herramienta para medir LOC eliminados, modificados y agregados, puede encontrar útil <a href="#">esta herramienta</a>).</li> <li>Ingrese estos datos en la <a href="#">plantilla de Estimación del tamaño</a>.</li> <li>Determine el tamaño total del programa</li> <li>Ingrese esta información en el formulario <a href="#">Resumen del plan del proyecto</a>.</li> </ul>
4	Hora	<ul style="list-style-type: none"> <li>Revise el <a href="#">registro de registro de tiempo</a> completo para ver si hay errores u omisiones.</li> <li>Usando su mejor recuerdo, corrija cualquier dato de tiempo faltante o incompleto.</li> </ul>
Criterio de salida	<ul style="list-style-type: none"> <li>Un programa completamente probado que cumple con los estándares de <a href="#">codificación</a> y conteo de tamaño</li> <li>Completadas <a href="#">revisión de diseño</a> y <a href="#">revisión de código listas de comprobación</a></li> <li>Completado la <a href="#">plantilla Informe de prueba</a></li> <li>Formulario de <a href="#">resumen del plan del proyecto</a> completado</li> <li>Formularios PIP completos que describen los problemas del proceso, sugerencias de mejora y lecciones aprendidas</li> <li>Registros de registro de <a href="#">tiempo</a> y <a href="#">defectos</a> completados</li> </ul>	

Para culminar este nivel se requiere de la revisión de la información ingresada en los distintos niveles anteriormente realizados estos son:

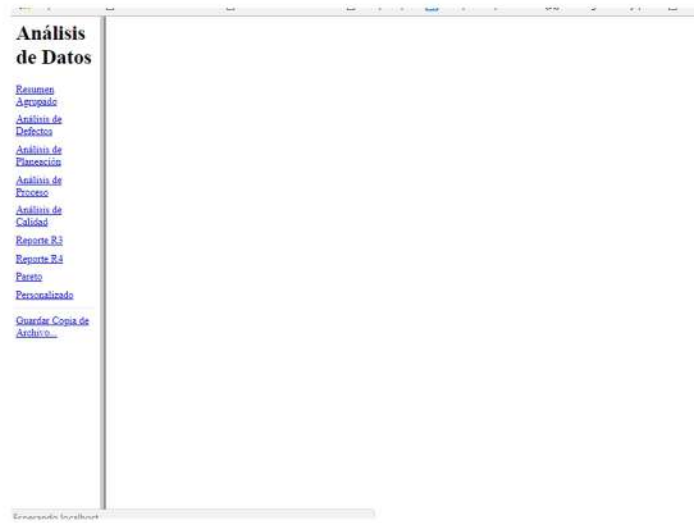
- Revisión de diseño y revisión de código listas de comprobación
- Plantilla Informe de prueba
- Formulario de resumen del plan de proyecto
- Formularios PIP
- Registros de registro de tiempo y defectos completados

Estas métricas obtenidas en PSP 2 se utilizan para mejorar la calidad del proceso en PSP 2.1 a través del establecimiento de verificación de tareas de diseño orientadas a mejorar las plantillas o formatos de diseño.

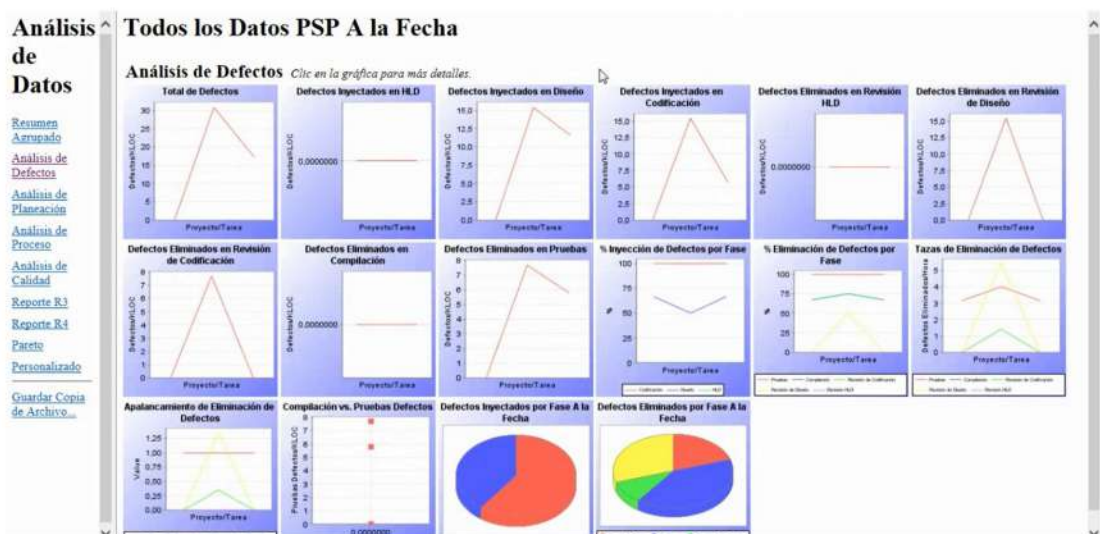
La metodología PSP es una metodología cíclica por lo tanto se debe de repetir los pasos de PSP0, PSP1 y PSP2 tantas veces como sea conveniente para mejorar la calidad del software en desarrollo.

### 4.3. Análisis de datos del PSP (PERSONAL SOFTWARE PROCESS)

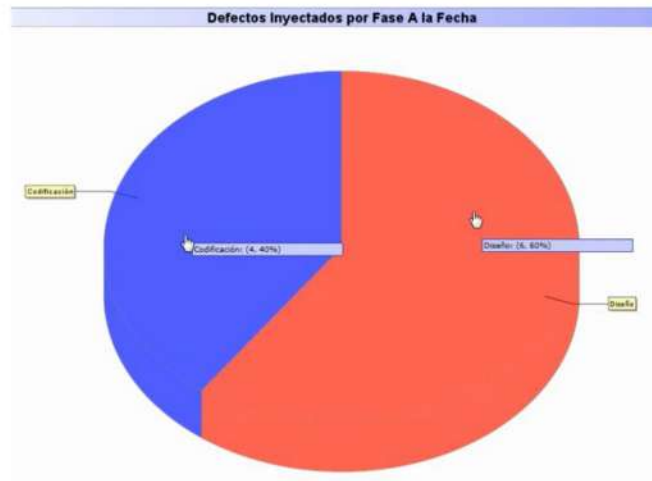
Para aplicar el Análisis de datos del PSP se utilizara la herramienta "Process Dashboard" en su versión 2.4.



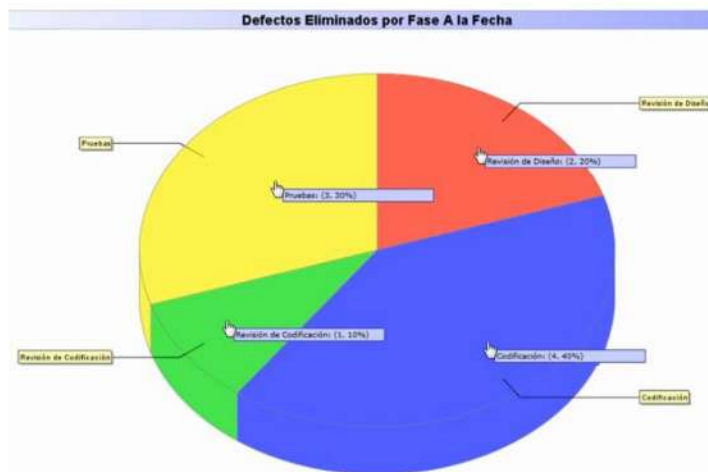
Accedemos a ANÁLISIS DE DEFECTOS



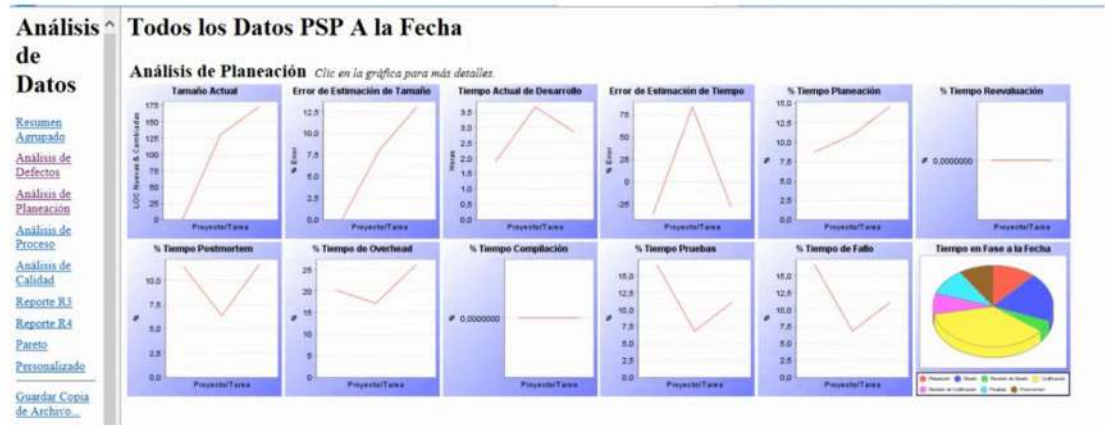
Accedemos al grafico de DEFECTOS INYECTADOS POR FASE HASTA LA FECHA



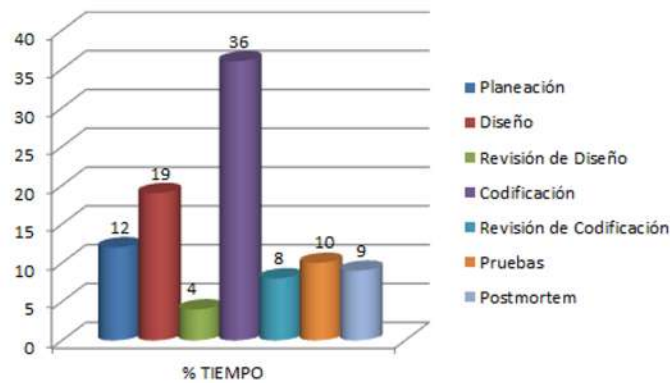
Accedemos al grafico de DEFECTOS ELIMINADOS POR FASE HASTA LA FECHA



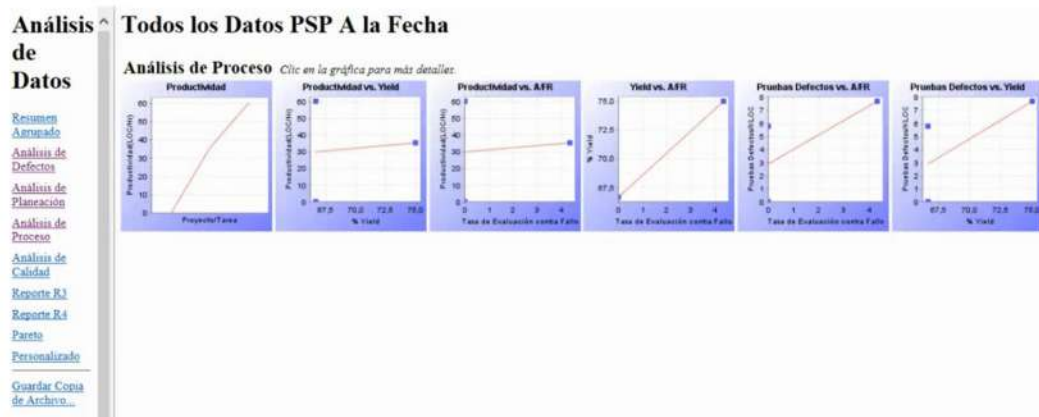
## Accedemos a ANÁLISIS DE PLANEACIÓN



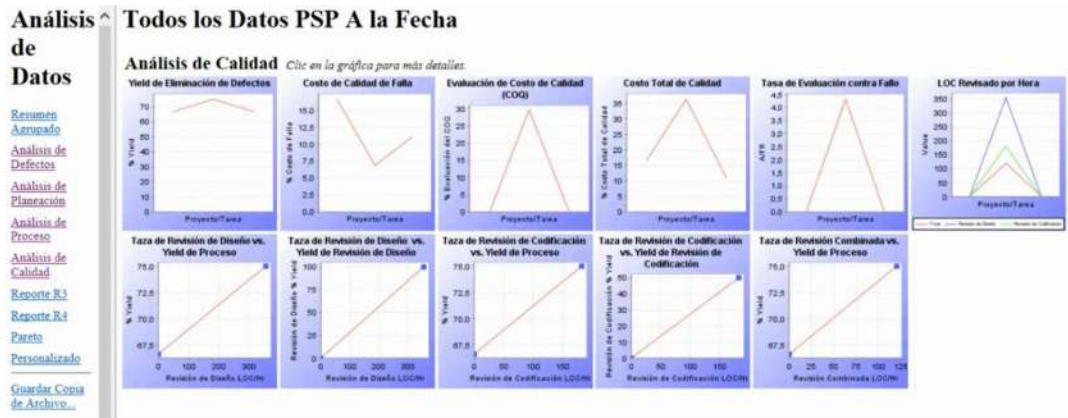
## Accedemos al grafico de TIEMPO EN FASE A LA FECHA



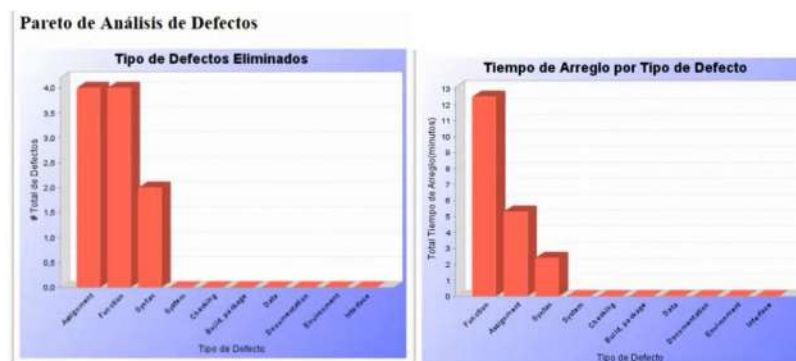
## Accedemos a ANÁLISIS DE PROCESO



## Accedemos a ANÁLISIS DE CALIDAD



Accedemos al GRÁFICOS DE PARETO este nos muestra el grafico de TIPOS DE DEFECTOS ELIMINADO además del grafico de TIEMPO DE ARREGLO POR TIPO DE DEFECTO



Finalmente conociendo las gráficas y los datos arrojados se determinó que la aplicación de la metodología PSP mejoró en gran medida el proceso de desarrollo del sistema de bolsa de trabajo de la Universidad de Huánuco, ya que permitió eliminar errores en cada fase del diseño, además se disminuyó los errores de estimación de tiempo de desarrollo, también nos permitió verificar el aumento de nuestra productividad en el desarrollo del software, por lo tanto se confirmó el beneficio del uso de la metodología PSP. Además de permitirnos construir nuestro propio proceso de desarrollo y como este proceso de desarrollo influye en la optimización del desarrollo del sistema. Con lo cual se cumple con los objetivos del presente trabajo de investigación.

## CAPITULO V

### DISCUSIÓN DE RESULTADOS

El propósito de esta investigación fue ver en qué medida la aplicación de la metodología PSP mejora el proceso de desarrollo del sistema de bolsa de trabajo.

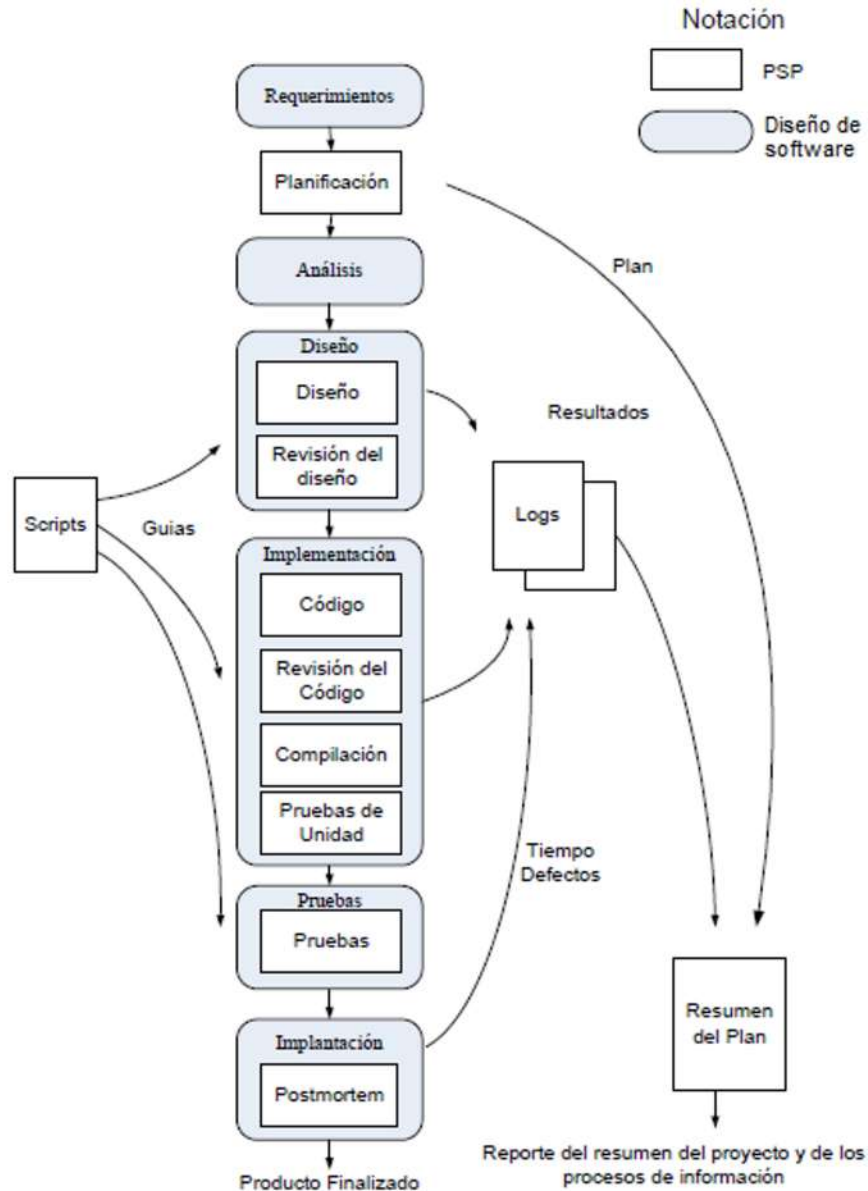
A partir de los resultados obtenidos en las investigaciones previas realizadas por Durán Hoyos (2015), Moreno Vallespir (2013), Dioses Farfán (2015) en Ecuador, Uruguay y Perú respectivamente; concordamos en que la aplicación del PSP incrementa la calidad de la producción del software, por lo tanto mejora el proceso de desarrollo del mismo.

Los resultados obtenidos por las investigaciones mencionadas anteriormente se obtuvieron con diferentes métodos de investigación además de otro tipo de herramientas entre ellas macro en Excel, document Student Workbook y hojas de cálculo. Es decir usaron diferente método de investigación, diferente tipo de herramienta pero las conclusiones son similares.

Este trabajo de investigación incrementa a los resultados encontrados la creación de una metodología personal que servirá para el diseño individual de futuros proyectos de diseño de software, para ello se consideró los flujos fundamentales de trabajo del diseño de software sobre los cuales se colocaron las mejores prácticas del PSP.

Estas fases o flujos son: requerimientos, planificación, diseño, implementación, pruebas, implantación.





A continuación se describe las fases de la metodología personal resultante del presente trabajo de investigación.

## REQUERIMIENTOS

La captura de requisitos o requerimientos sirve para ordenar el desarrollo hacia el sistema correcto. Esto se alcanza mediante una buena descripción de los requisitos del sistema, para que se pueda llegar a un acuerdo entre el cliente y el desarrollador sobre qué debe y qué no debe hacer el sistema.

Para este flujo de trabajo se recomienda:

- Producir un informe de los requerimientos del programa.

- Asegurar que el informe escrito de requerimientos sea claro y sin ambigüedad.
- Resolver cualquier pregunta.

## **PLANIFICACIÓN**

El objetivo de realizarlo es adquirir una comprensión mucho más precisa de todos y cada uno de los requisitos y una mayor descripción de los mismos que sea más fácil de mantener y que ayude a estructurar el sistema completo.

Elaborar las siguientes prácticas:

- Diseño conceptual del sistema.
- Estimación de Tamaño.
- Estrategia del ciclo de desarrollo.
- Estimación de Recursos.
- Planificación de tareas y Horarios.
- Estimación de Defectos.

## **DISEÑO**

En el diseño se modela el sistema y se define su forma para que soporte todos los requisitos.

Obtener la siguiente información:

- Especificaciones externas.
- Módulo de Diseño.
- Prototipos.
- Estrategia de desarrollo.
- Log de rastreo de problemas.
- El diseño global del programa, incluidas las plantillas de especificaciones de control principal y de rutina.

Realizar la revisión del diseño que contiene la siguiente información:

- Alcance del diseño.
- Verificación lógica.

- Verificación de reuso.
- Verificación de la estrategia de Desarrollo.
- Reparar Defectos.

## **IMPLEMENTACIÓN**

Cumplir con los siguientes objetivos:

- Definir el código de organización, en términos de implementación de Subsistemas
- Implementar clases y objetos en términos de componentes
- Probar el desarrollo de los componentes como unidades
- Integrar los resultados producidos de las implementaciones individuales
- Revisión del diseño.
- Revisión de la Codificación.
- Compilación.
- Reasegurar y reciclar.

## **PRUEBAS**

El resultado principal es el modelo de prueba, el cual describe cómo ha sido probado el sistema.

Cumplir con los siguientes objetivos:

- Probar los módulos hasta que la prueba se ejecute sin errores.
- Reparar todos los defectos establecidos.
- Registrar los defectos en el Log de registro de defectos.
- Registrar el tiempo en el Log de registro del tiempo.
- Completar la plantilla de reporte de pruebas en las pruebas dirigidas y en los resultados obtenidos.

## **IMPLANTACIÓN**

El propósito principal de este flujo de trabajo es producir una versión del producto y entregar el software al usuario final, esto incluye las siguientes actividades:

- Producir una versión del software.

- Empaquetar el software.
- Distribuir el software.
- Instalación del software.
- Proveer de ayuda y asistencia a los usuarios.

Obtener la siguiente información:

- Defectos inyectados.
- Defectos removidos.
- Tamaño.
- Tiempo.
- Completar la forma del plan del proyecto con el tiempo actual, defectos, tamaño de los datos actualizados.
- Completar la forma del resumen del ciclo con datos actuales del ciclo.

Siguiendo la metodología individual se pudo desarrollar el software BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO (anexo 3) incrementando su calidad y disminuyendo el tiempo de desarrollo.

## CONCLUSIONES

- La aplicación del PSP cumple con el principal objetivo del presente trabajo de investigación ya que incidió de gran manera en la mejora del proceso de desarrollo del sistema de bolsa de trabajo de la Universidad de Huánuco pues ayudó a administrar el tiempo y cumplir con los compromisos de desarrollo.
- La aplicación del PSP nos facilita la construcción de nuestro propio proceso de desarrollo ya que nos permitió: medir y prevenir defectos, conocer cuánto y en qué tiempo se escribe el código fuente del sistema, llevar un registro sobre las pruebas efectuadas, saber puntualmente que se probó y cuál fue su resultado y además permitió saber que etapas del desarrollo del software son prescindibles o imprescindibles.
- La aplicación del PSP optimizó el desarrollo del software de bolsa de trabajo de la Universidad de Huánuco ya que nos permite realizar mejores estimaciones y planes de trabajo por lo tanto nos ahorra tiempo y recursos.

## RECOMENDACIONES

- Se recomienda al desarrollador ampliar sus capacidades relacionadas con procesos de calidad de software a nivel personal.
- Que la adopción del PSP sea de manera progresiva a medida que se vayan asimilando y desarrollando las prácticas, habilidades y la documentación que el PSP sugiere.
- Una vez empleada la mejora en los procesos del desarrollo de software antes mencionados se recomienda implementarla en nuevos proyectos para alcanzar un nivel cada vez más alto de calidad en el desarrollo de software.

## REFERENCIAS BIBLIOGRÁFICAS

- Sánchez Carlessi H. y Reyes Meza C. (2006). Metodología y diseños en investigación científica. Edit. Visión Universitaria. Lima – Perú.
- Humphrey, W. S. (1996). Using a defined and measured personal software process. IEEE software, 13(3), 77-88.
- Roberto Miñana (2016) Recuperado el 26 de Febrero de 2017, de <http://calidadyssoftware.blogspot.com/2011/09/metricas-loc.html>.
- ISO. (2011). Qué es ISO 9001:2008?, Recuperado el 25 de Febrero de 2017, de <http://www.normas9000.com/que-es-iso-9000.html>.
- Eduardo Corrales, REINGENIERA DE PROCESOS, Recuperado el 2 de Enero de 2017, <http://www.slideshare.net/jcfdezmx2/reingenieria-de-procesos-presentation>.
- Carlos Javier Pérez Escobar (2015), Recuperado el 2 de Enero de 2017, <http://asprotech.blogspot.pe/2010/06/personal-software-process-psp-elementos.html>
- Jesús Antonio Solís García (2012), Recuperado el 3 de Enero de 2017, <https://jjegonzalezf.wordpress.com/2010/12/14/psp-personal-software-process-un-enfoque-practico/>
- Moisés Salinas Silva (2013), Recuperado el 3 de Enero de 2017, <https://prezi.com/7ztatvpdenom/modelos-tsp-psp/>
- Hayes, William; & Over, James. Personal Software Process (PSP): An Empirical Study of the Impact of PSP on Individual Engineers (CMU/SEI-97-TR-001). Software Engineering Institute, Carnegie Mellon University, 1997. Recuperado el 15 de junio de 2016, <http://www.sei.cmu.edu/library/abstracts/reports/97tr001.cfm>
- Kusakabe, Shigeru; Omori, Yoichi; and Araki, Keijiro. "A Combination of a Formal Method and PSP for Improving Software Process: An Initial Report," 67-75. TSP Symposium 2012 Proceedings (CMU/SEI-2012-SR-015). Software Engineering Institute, Carnegie Mellon University, 2012. Recuperado el 7 de setiembre de 2016, <http://www.sei.cmu.edu/library/abstracts/reports/12sr015.cfm>

# ANEXOS

## ANEXO 1: ESTÁNDARES DE CODIFICACIÓN

ESTANDAR DE CODIFICACION	
<b>Propósito</b>	Este documento tiene como propósito guiar la implementación de los programas que desarrollare utilizando el lenguaje de programación java
<b>Conteo Estándar</b>	<ul style="list-style-type: none"> <li>• Cada línea física representa una línea de código.</li> <li>• No se cuentan líneas que solo contienen espacios en blanco ni líneas correspondientes a documentación.</li> <li>• No se contarán paquetes.</li> <li>• No se tendrá en cuenta líneas autogeneradas por el entorno.</li> </ul>
<b>Campos, variables.</b>	<ul style="list-style-type: none"> <li>• Los nombres de campos y variables iniciaran con letra minúscula.</li> <li>• Si tienen nombre compuesto este debe ir sin espacios y la palabra que lo compone debe comenzar con caracteres en mayúscula.</li> <li>• Los nombres no deben contener caracteres símbolos.</li> <li>• No deben empezar por carácter número.</li> <li>• Los nombres deben contener como mínimo 3 caracteres y estos deben dar información de lo que contendrá el campo.</li> </ul>
<b>Interfaces, estructuras, clases</b>	<ul style="list-style-type: none"> <li>• Debe ajustarse a las reglas de nombres definidos para las variables con la diferencia que los nombres de estos deben iniciar con letra mayúscula.</li> <li>• El nombre debe expresar de manera breve la funcionalidad de la clase.</li> <li>• Tendrá un encabezado distintivo "\$Clase": para la clase. Y después de la llave que cierra la clase tendrá una Línea de documentación <code>//\$endClass</code> para identificar el final de la clase.</li> <li>• Tendrá un encabezado distintivo "\$Estruc": para la Estructura.</li> <li>• Tendrá un encabezado distintivo "\$Inter": para la interface.</li> </ul>
<b>Bloques de código</b>	Deben tener una sangría de mínimo dos espacios para facilitar la interpretación del código en las líneas que conforman un bloque.
<b>Comentarios</b>	
<b>De Línea</b>	<ul style="list-style-type: none"> <li>• Se utilizara una línea por comentario y esta podrá o no finalizar en punto.</li> <li>• Se debe prestar atención a la ortografía.</li> <li>• Se usaran para describir el funcionamiento de variables, métodos y clases.</li> </ul>
<b>De bloque</b>	<ul style="list-style-type: none"> <li>• Iniciará con <code>/*</code>.</li> <li>• En los intermedios iniciara con <code>*</code>.</li> <li>• Terminará con <code>*/</code></li> <li>• Se utilizará una línea por comentario y esta podrá o no finalizar con punto.</li> <li>• Se utilizará para el encabezado el cual contendrá nombre del programa, fecha, nombre de quien lo realiza y una breve descripción.</li> </ul>
<b>Métodos o funciones</b>	<ul style="list-style-type: none"> <li>• Los nombres se ajustaran a las reglas definidas para nombrar clases.</li> <li>• El nombre debe dar una breve descripción de lo que hace.</li> <li>• Tendrá un encabezado distintivo "\$Metod": para la clase.</li> <li>• Tendrá una línea de comentario en la línea siguiente de la llave de cierre para identificación <code>//\$endMetod</code>.</li> </ul>
<b>Sentencias de decisión</b>	<ul style="list-style-type: none"> <li>• Utilizar llaves aunque solo contenga una línea.</li> <li>• Utilizar una sintaxis explicativa para los campos que están siendo objeto de comparación</li> </ul>
<b>Sentencias de Iteración</b>	<ul style="list-style-type: none"> <li>• Documentar las sentencias que generan ruptura.</li> </ul>



## ANEXO 2: DOCUMENTOS PSP

### 1) PSP2 Design Review Checklist

Modulo \_\_\_\_\_ Fecha \_\_\_\_\_  
 Responsable \_\_\_\_\_ Language \_\_\_\_\_

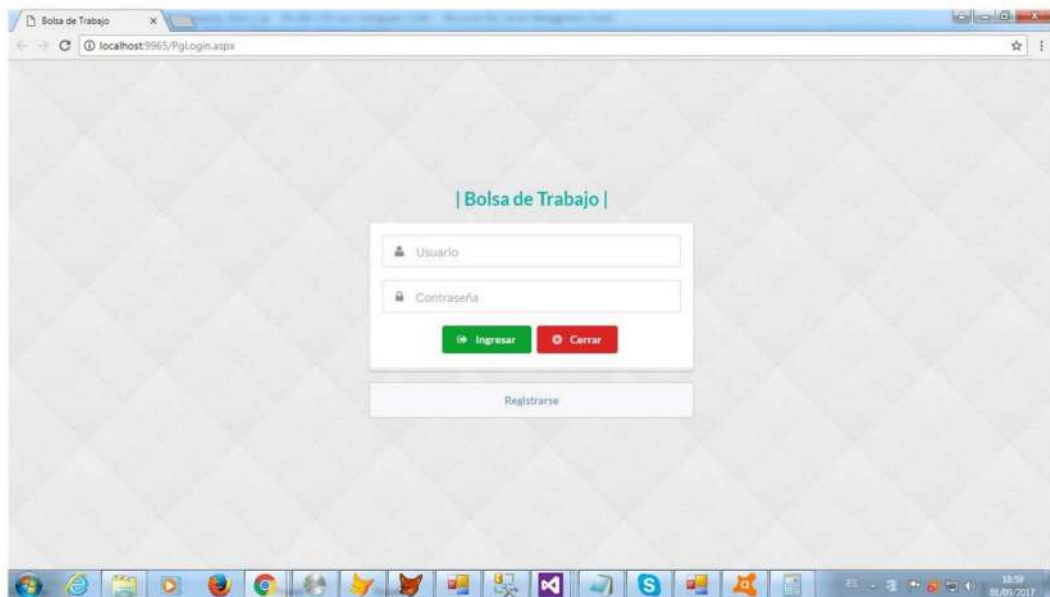
<b>Propósito General</b>	Para guiarlo en la realización de una revisión de diseño efectiva				
	<ul style="list-style-type: none"> <li>- Revise el programa completo para cada categoría de lista de verificación; ¡no intente revisar por más de una categoría a la vez!</li> <li>- A medida que completa cada paso de revisión, marque ese elemento en el cuadro de la derecha.</li> <li>- Complete la lista de verificación para un programa o unidad de programa antes de revisar la próxima.</li> </ul>				
Completar	Verificar que el diseño cubra todos los requisitos aplicables. <ul style="list-style-type: none"> <li>- Todos los productos especificados son producidos.</li> <li>- Todas las entradas necesarias están amuebladas.</li> <li>- Se incluyen todos los requisitos necesarios.</li> </ul>				
Límites externos	Cuando el diseño asuma o dependa de límites externos, determine si el comportamiento es correcto a valores nominales, límites y más allá de los límites.				
Lógica	<ul style="list-style-type: none"> <li>- Verifique que la secuencia del programa sea correcta.               <ul style="list-style-type: none"> <li>Las pilas, listas, etc. están en el orden correcto.</li> <li>La recursión se desenrolla apropiadamente.</li> </ul> </li> <li>- Verifique que todos los bucles se inicien, incrementen y finalicen correctamente.</li> <li>- Examine cada enunciado condicional y verifique todos los casos.</li> </ul>				
Límites internos	Cuando el diseño asuma o dependa de límites internos, determine si el comportamiento es correcto a valores nominales, límites y más allá de los límites.				
Casos especiales	<ul style="list-style-type: none"> <li>- Verifica todos los casos especiales.</li> <li>- Asegure el funcionamiento correcto con valores vacíos, completos, mínimos, máximos, negativos y ero para todas las variables.</li> <li>- Proteger contra condiciones fuera de los límites, desbordamiento y subdesbordamiento.</li> <li>- Asegurar que las condiciones "imposibles" sean absolutamente imposibles.</li> <li>- Manejar todas las posibles condiciones incorrectas o de error.</li> </ul>				
Uso Funcional	<ul style="list-style-type: none"> <li>- Verifique que todas las funciones, procedimientos o métodos se entiendan por completo y se usen correctamente.</li> <li>- Verifique que todas las abstracciones con referencia externa estén definidas con precisión.</li> </ul>				
Consideraciones del sistema	<ul style="list-style-type: none"> <li>- Verifique que el programa no cause que se excedan los límites del sistema.</li> <li>- Verifique que todos los datos confidenciales provengan de fuentes confiables.</li> <li>- Verifique que todas las condiciones de seguridad cumplan con las especificaciones de seguridad.</li> </ul>				
Nombres	Comprueba eso <ul style="list-style-type: none"> <li>- todos los nombres especiales son claros, definidos y autenticados</li> <li>- los alcances de todas las variables y parámetros son evidentes por sí mismos o definidos</li> <li>- todos los artículos con nombre se usan dentro de sus ámbitos declarados</li> </ul>				
Estándares	Asegúrese de que el diseño cumpla con todos los estándares de diseño aplicables.				

## 2) PS2 Code Review Checklist

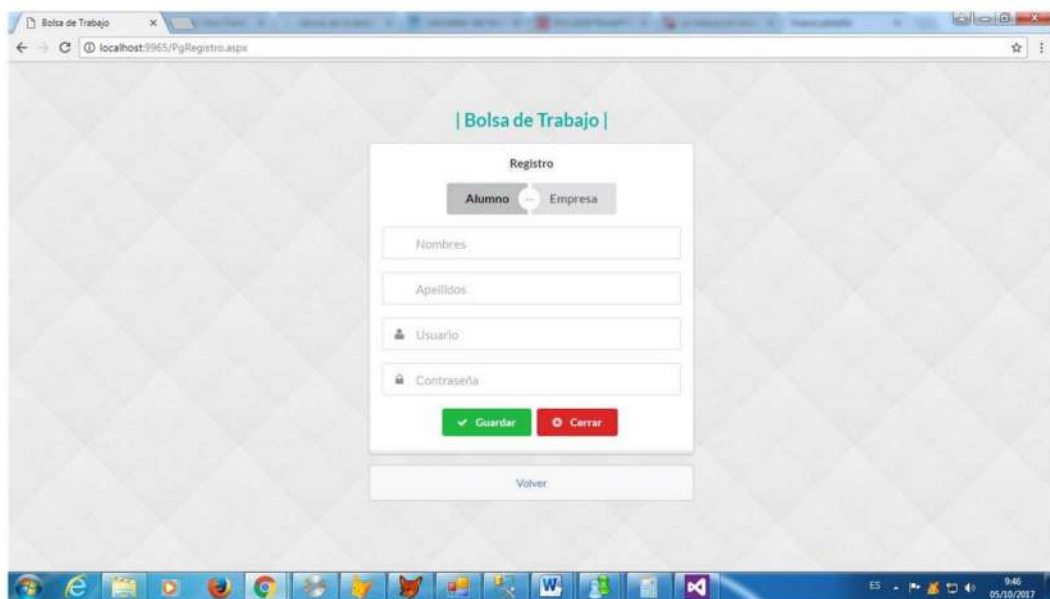
Modulo \_\_\_\_\_ Fecha \_\_\_\_\_  
 Responsable \_\_\_\_\_ Language \_\_\_\_\_

<b>Propósito</b>	Para guiarlo en la realización de una revisión de código efectiva				
<b>General</b>	<ul style="list-style-type: none"> <li>- Revise el programa completo para cada categoría de lista de verificación; ¡no intente revisar por más de una categoría a la vez!</li> <li>- A medida que completa cada paso de revisión, marque ese elemento en el cuadro de la derecha.</li> <li>- Complete la lista de verificación para un programa o unidad de programa antes de revisar el siguiente.</li> </ul>				
Completar	Verifique que el código cubre todo el diseño.				
Incluye	Verifique que las inclusiones estén completas.				
Inicialización	Verifique la inicialización de variables y parámetros. <ul style="list-style-type: none"> <li>- al inicio del programa</li> <li>- al inicio de cada ciclo</li> <li>- en la entrada clase / función / procedimiento</li> </ul>				
Llamadas	Compruebe los formatos de llamada de función. <ul style="list-style-type: none"> <li>- punteros</li> <li>- parámetros</li> <li>- uso de '&amp;'</li> </ul>				
Nombres	Verifique la ortografía y el uso del nombre. <ul style="list-style-type: none"> <li>- ¿Es consistente?</li> <li>- ¿Está dentro del alcance declarado?</li> <li>- ¿Todas las estructuras y clases usan referencia '!'?</li> </ul>				
Strings	Verifique que todas las cadenas sean <ul style="list-style-type: none"> <li>- identificado por punteros</li> <li>- terminado por NULL</li> </ul>				
Punteros	Compruebe que todo <ul style="list-style-type: none"> <li>- los punteros se inicializan NULL</li> <li>- los punteros se eliminan solo después de nuevo</li> <li>- los nuevos punteros siempre se eliminan después del uso</li> </ul>				
Formato de salida	Verifica el formato de salida. <ul style="list-style-type: none"> <li>- El paso de línea es apropiado.</li> <li>- El espaciado es correcto.</li> </ul>				
() Pairs	Asegúrese de que () sean correctos y coincidan.				
Operadores Lógicos	<ul style="list-style-type: none"> <li>- Verifique el uso correcto de ==, =,   , y así sucesivamente.</li> <li>- Verifica cada función lógica para ().</li> </ul>				
Comprobación línea por línea	Verifique cada línea de código para <ul style="list-style-type: none"> <li>- sintaxis de instrucciones</li> <li>- puntuación adecuada</li> </ul>				
Estándares	Asegúrese de que el código cumpla con los estándares de codificación.				
Archivo abierto y cerrado	Verifique que todos los archivos sean <ul style="list-style-type: none"> <li>- debidamente declarado</li> <li>- abrió</li> <li>- cerrado</li> </ul>				

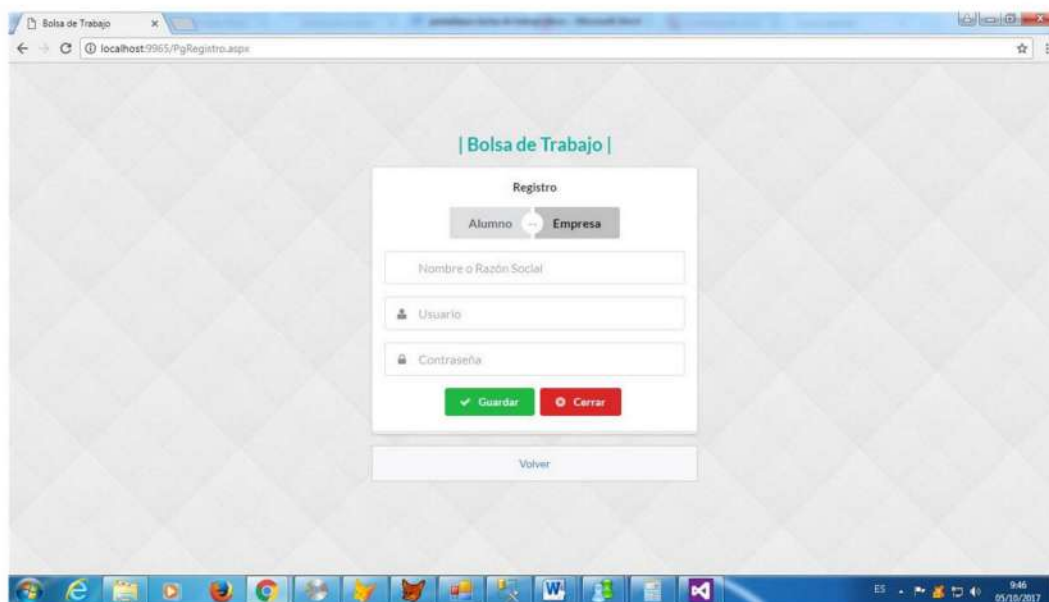
## ANEXO 3: SOFTWARE BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO



Login del sistema donde se ingresa el usuario y la contraseña. También hay una opción para registrarse en caso no se tengo el usuario y la contraseña.

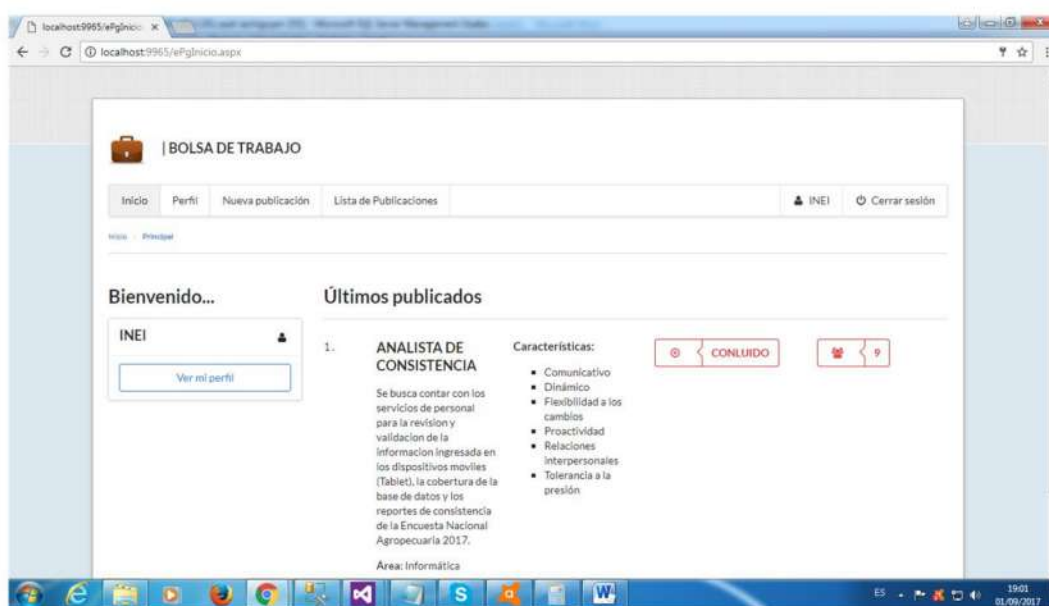


Portal de Registro de los alumnos.

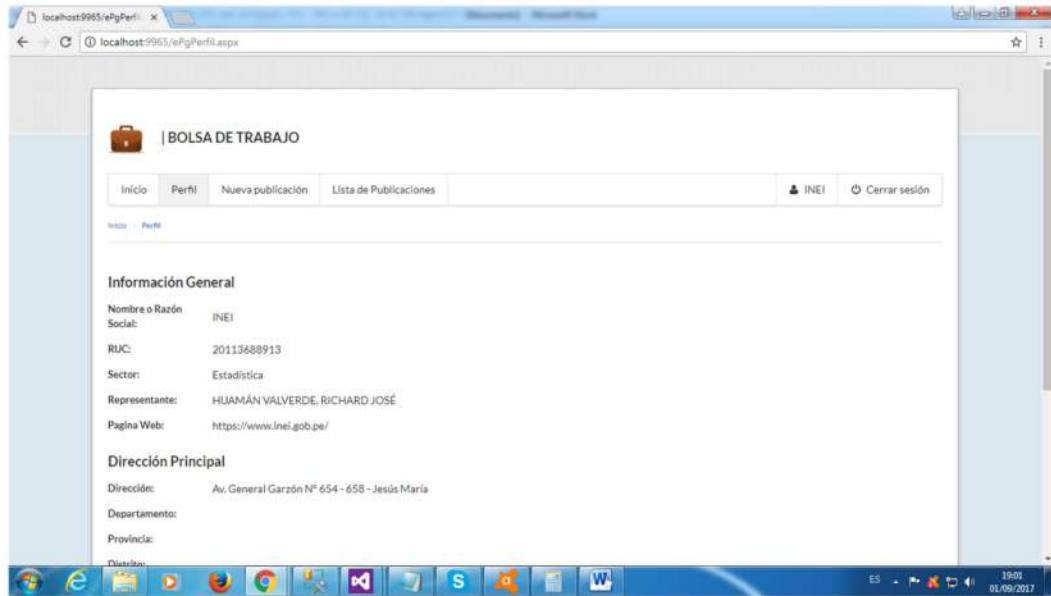


Portal de registro de las empresas.

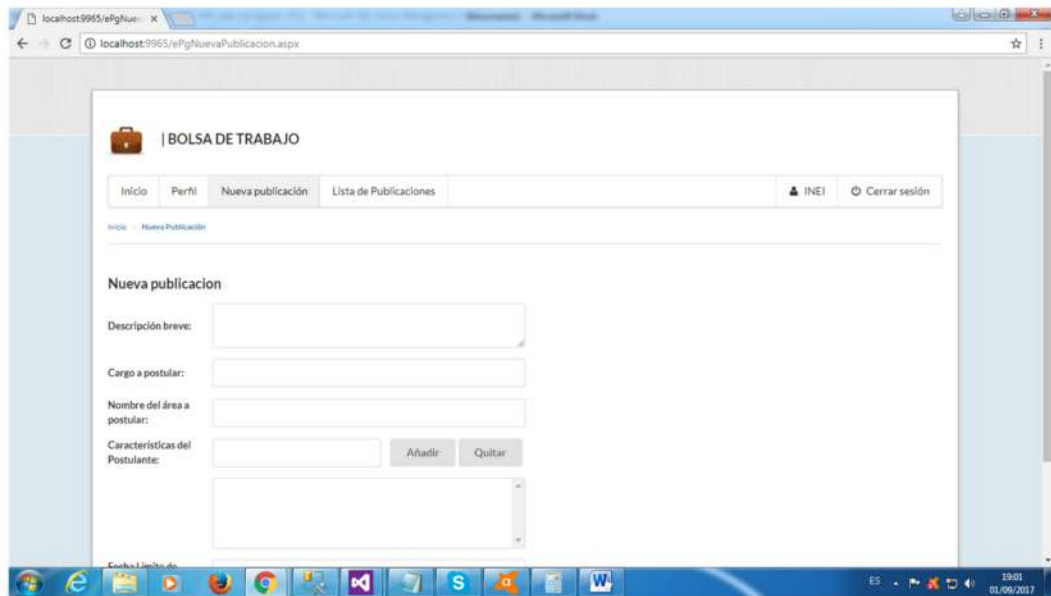
EMPRESA:



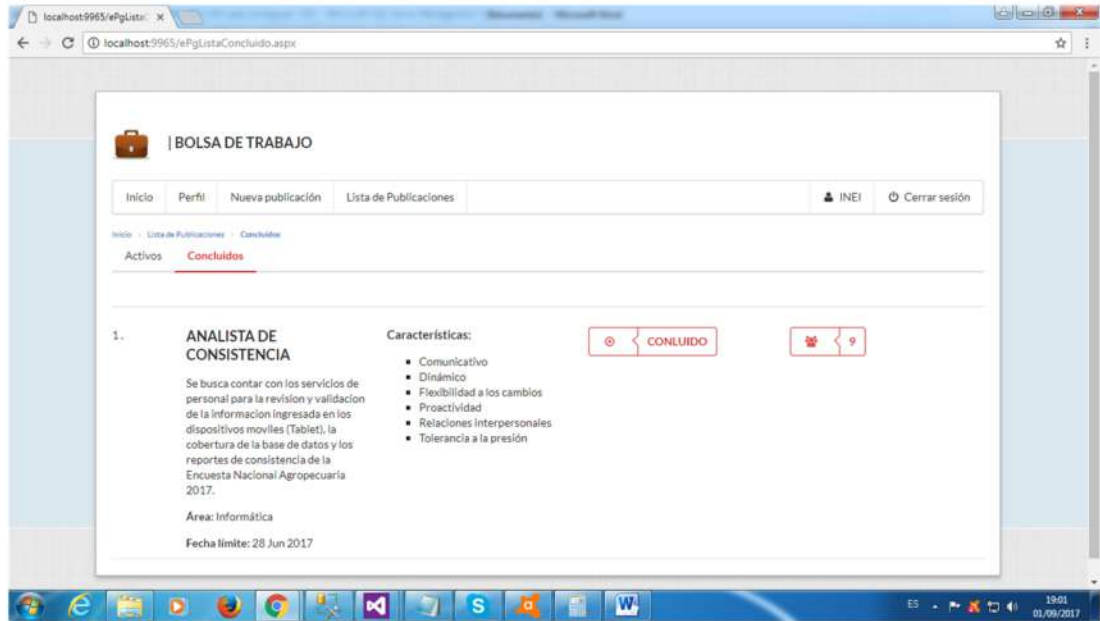
Ventana de inicio de la empresa, se muestra los 5 últimos avisos que han sido publicados. Se muestra el estado y la cantidad de postulantes.



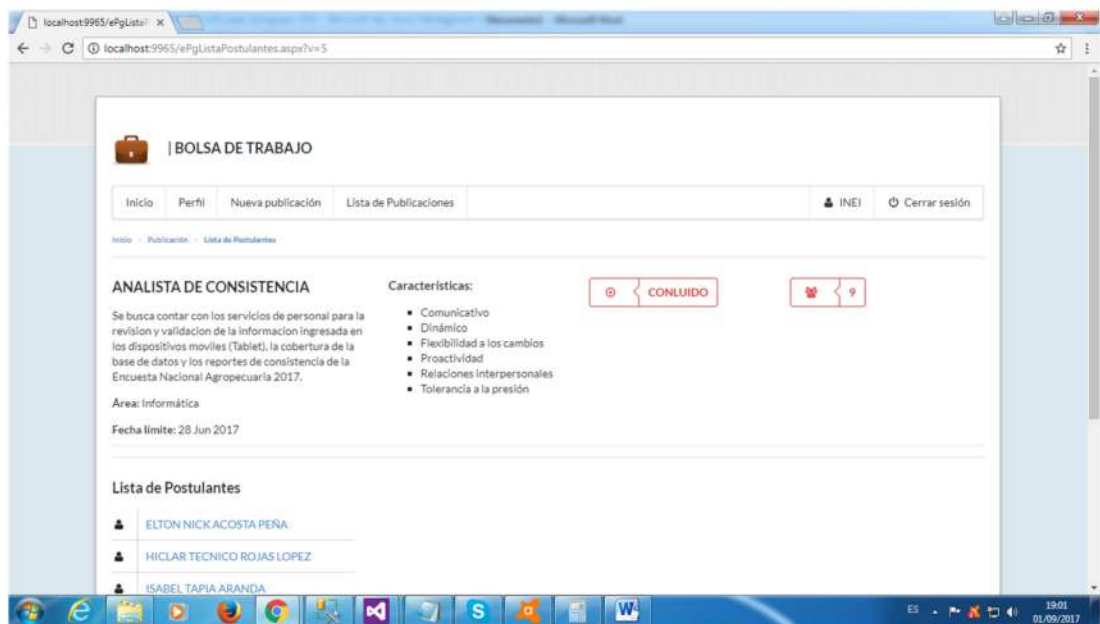
Ventana del perfil de la empresa. Están todos los datos y tiene la opción de editar.



Ventana para registrar una nueva publicación. Se define la fecha límite. De modo que si ya se pasa dicha fecha la publicación pasa de estado activo ha concluido.

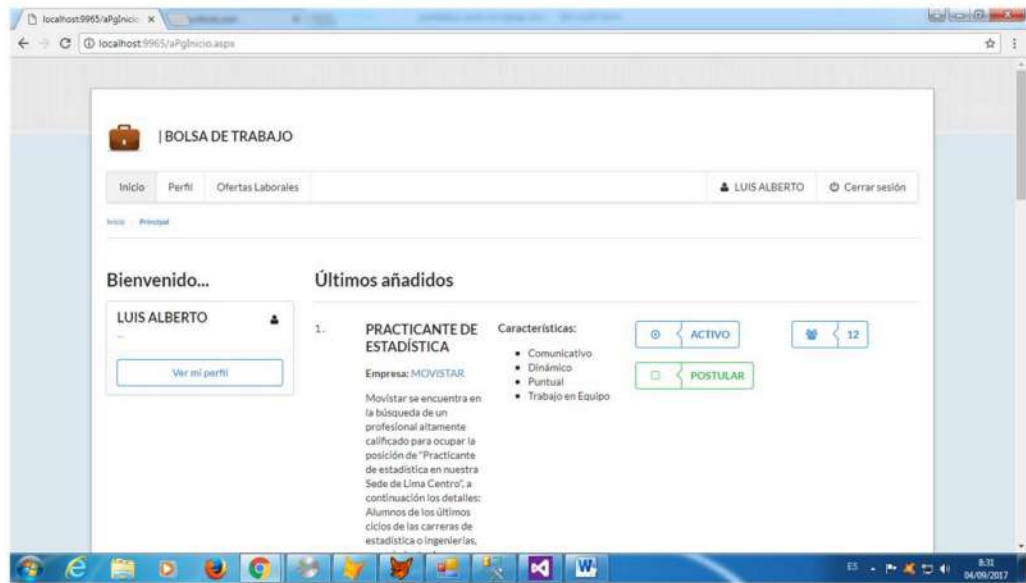


Lista de Publicaciones, tanto activos como concluidos.

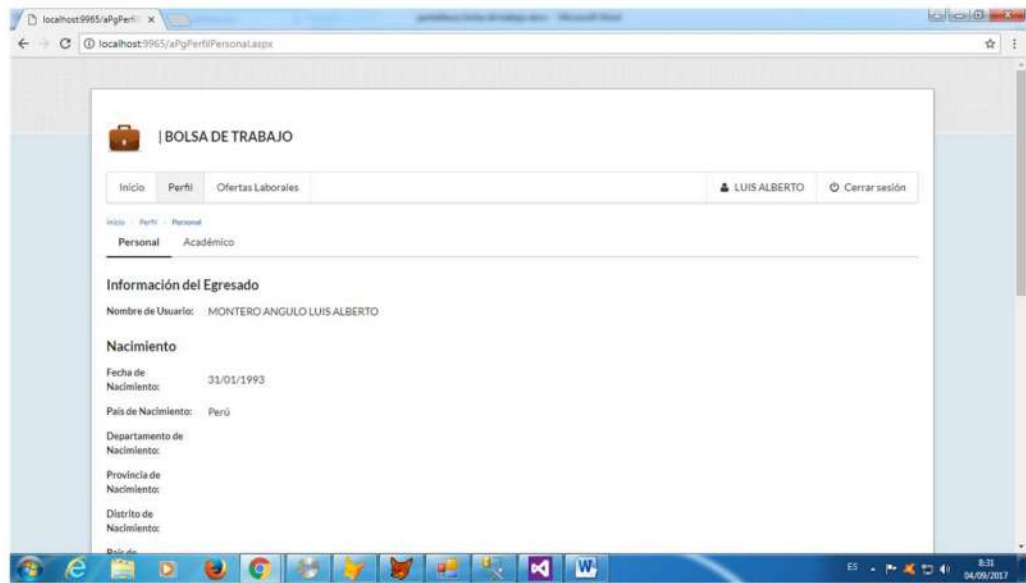


Al hacer clic en el número de postulantes de muestra más a detalle la publicación como el nombre de cada postulante con la opción de ver el perfil de cada uno de ellos.

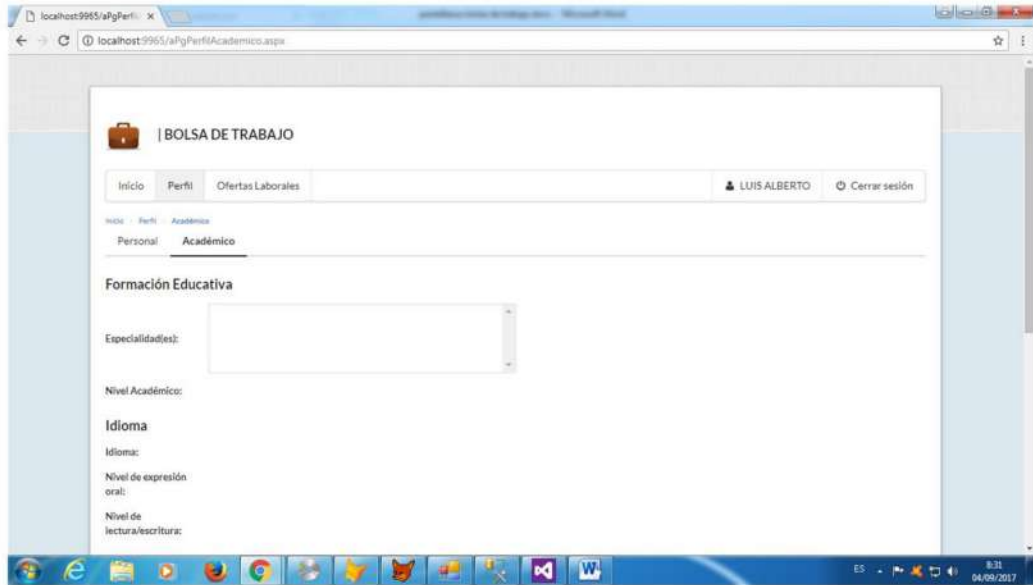
ALUMNO:



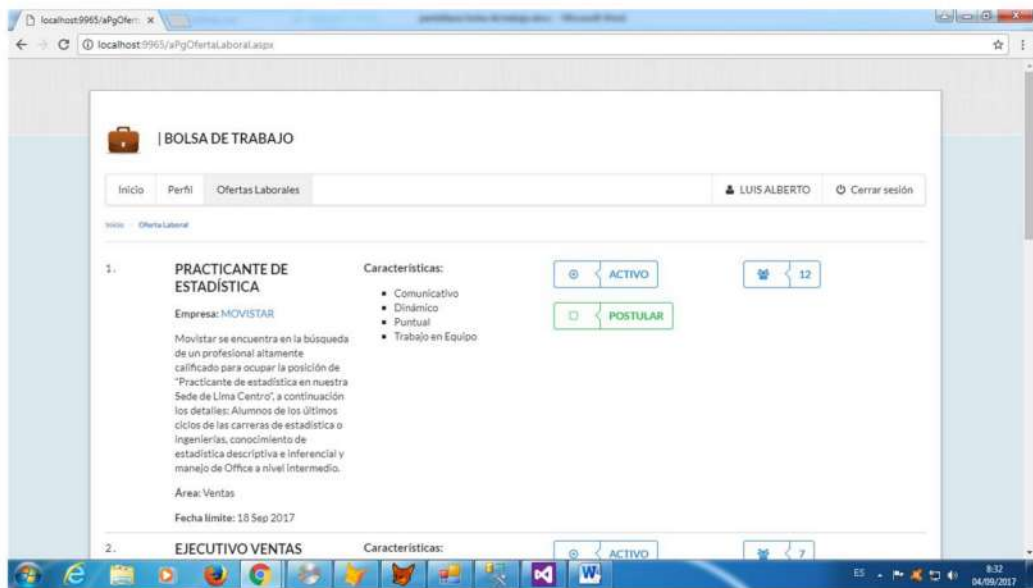
Ventana de inicio del alumno, se visualiza las 5 últimas publicaciones añadidas de todas las empresas.



Ventana de perfil personal del alumno. Se puede editar todos los campos.

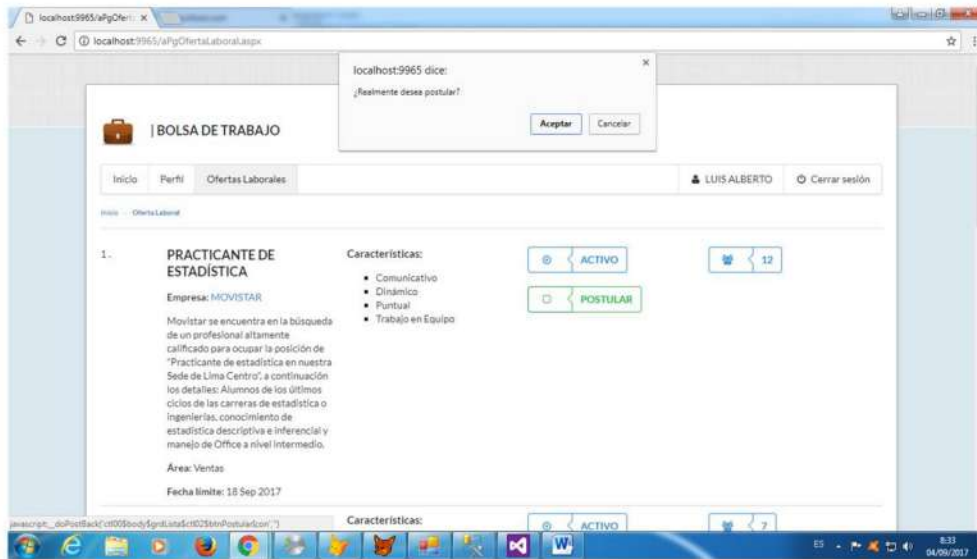


Ventana de perfil Académico del alumno. Se puede editar todos los campos.

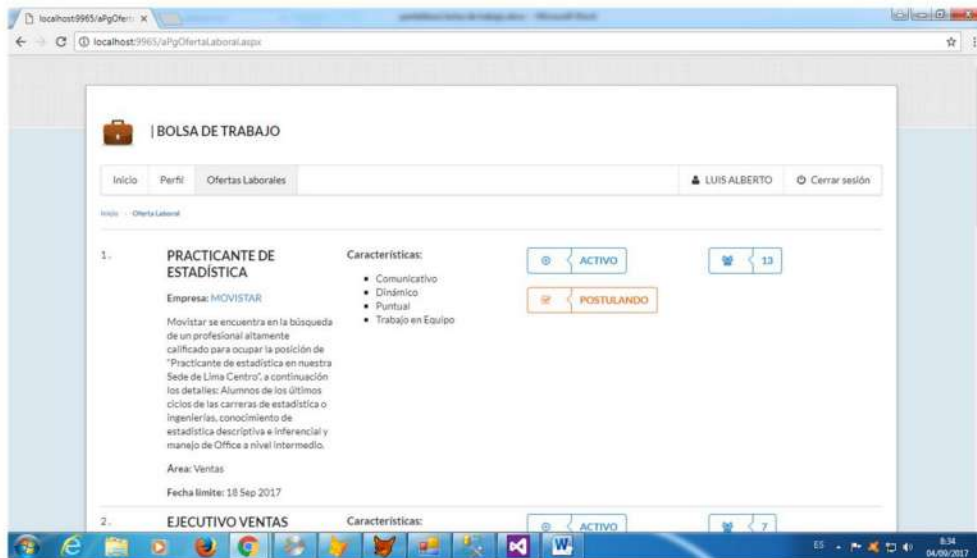


Lista de publicaciones activas, ordenadas por fecha.

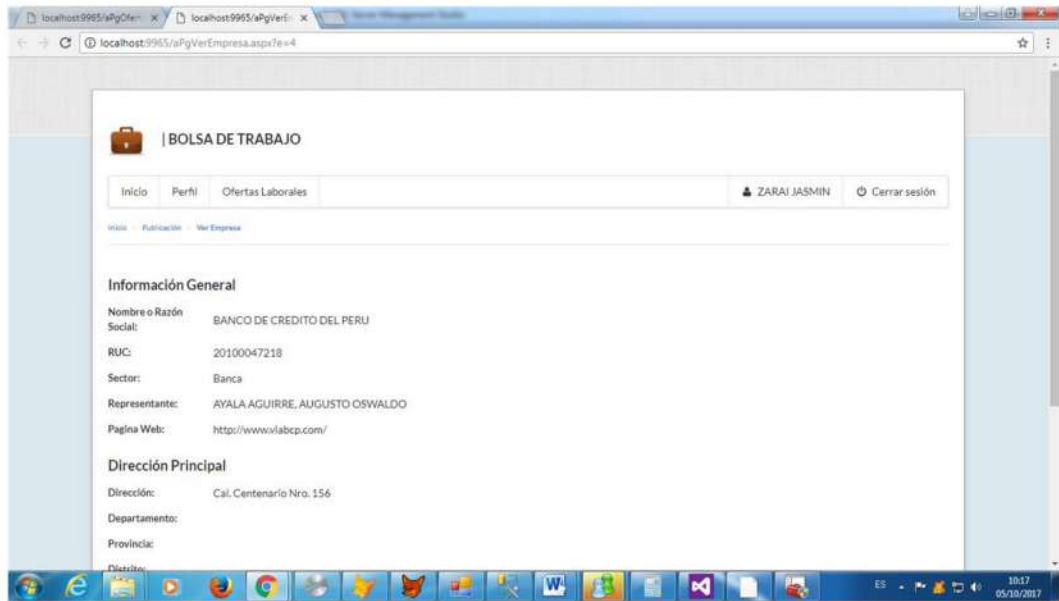




Para postular solo se debe hacer clic en el recuadro verde.



Visualización de una publicación a la cual se está postulando.



Visualización de la empresa desde el portal del alumno.

## ANEXO 4: MATRIZ DE CONSISTENCIA

Título de la Investigación:

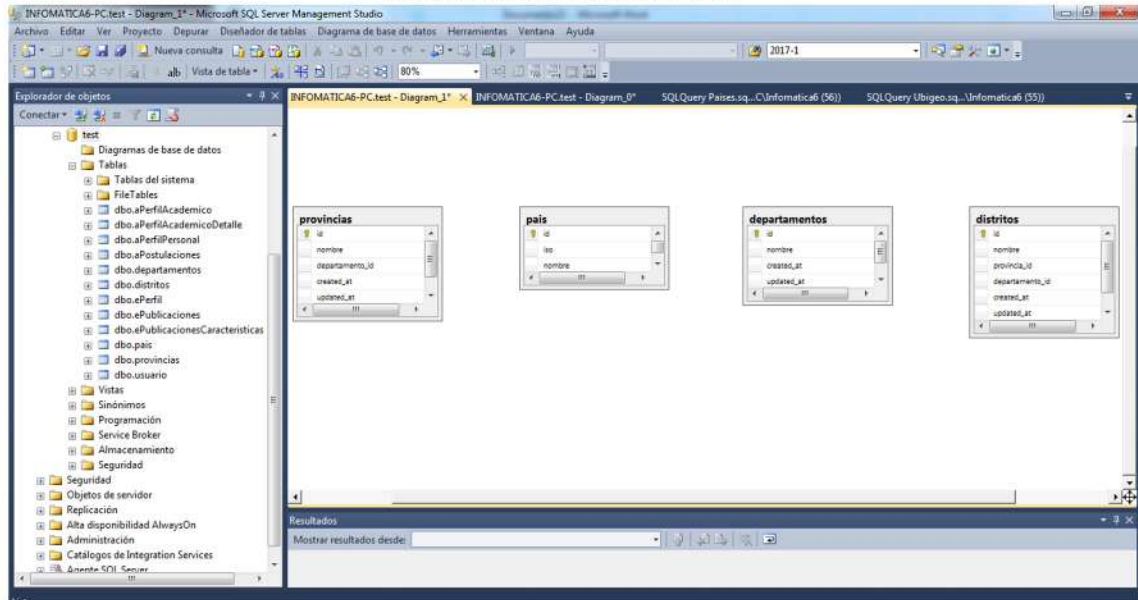
“Aplicación de PSP (Personal Software Process) para el desarrollo del sistema de bolsa de trabajo de la Universidad de Huánuco - 2016”

FORMULACION DEL PROBLEMA	PROBLEMA	OBJETIVOS	VARIABLES	DIMENSIONES	INDICADORES	METODOLOGIA		
<p>APLICACIÓN DE PSP (PERSONAL SOFTWARE PROCESS) PARA EL DESARROLLO DEL SISTEMA DE BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO - 2016</p>	<p><b>PROBLEMA GENERAL</b></p> <p>¿En qué medida la aplicación de la metodología PSP mejora el proceso de desarrollo del sistema de bolsa de trabajo de la Universidad de Huánuco?</p>	<p><b>OBJETIVO GENERAL</b></p> <p>Determinar en qué medida la aplicación de la metodología PSP mejorara el proceso de desarrollo del sistema de bolsa de trabajo de la Universidad de Huánuco.</p>	<p><b>VARIABLE INDEPENDIENTE</b></p> <p>Aplicación del proceso de software personal</p>	<p><b>MEDIDAS DE CALIDAD</b></p>	Densidad del defecto	<p>Tipo aplicativo y tecnológico porque utilizamos la teoría para desarrollar un producto software utilizando un nuevo enfoque en el proceso.</p>		
	<p><b>PROBLEMAS ESPECÍFICOS</b></p> <p>¿Cómo beneficiará el uso de la metodología PSP al construir nuestro propio proceso de desarrollo?</p> <p>¿Cómo influenciará la aplicación de la metodología PSP en la optimización del desarrollo del software de bolsa de trabajo de la Universidad de Huánuco?</p>	<p><b>OBJETIVOS ESPECÍFICOS</b></p> <p>Determinar el beneficio del uso de la metodología PSP al construir nuestro propio proceso de desarrollo.</p> <p>Determinar la influencia de la aplicación de la metodología PSP en la optimización del desarrollo del software de bolsa de trabajo de la Universidad de Huánuco.</p>			<p><b>VARIABLE DEPENDIENTE</b></p> <p>Desarrollo del sistema de bolsa de trabajo</p>		<p><b>MEDIDAS DE CALIDAD</b></p>	Razón de revisión
								Razón del tiempo de desarrollo
								Razón del defecto
			Productividad					
							Defectos por hora	
							Defectos renovables viables	
							Nivelación de la razón de falla	

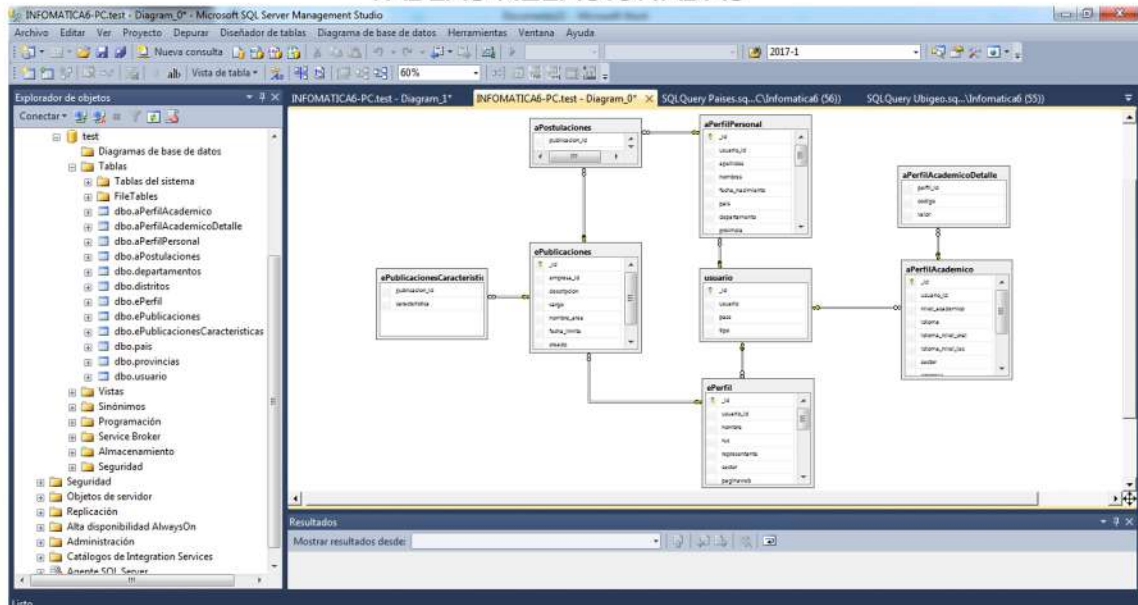
## ANEXO 5: EVIDENCIAS DE APLICACIÓN Y DESARROLLO

### EVIDENCIAS DE LA CREACIÓN DE BASE DE DATOS

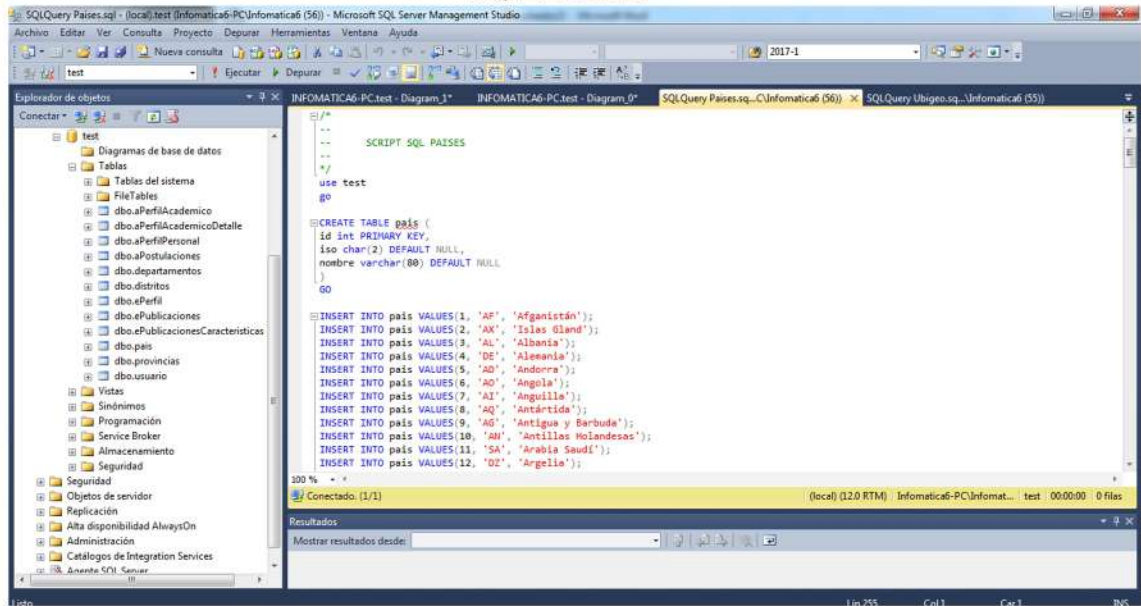
#### TABLAS INDIVIDUALES



#### TABLAS RELACIONADAS



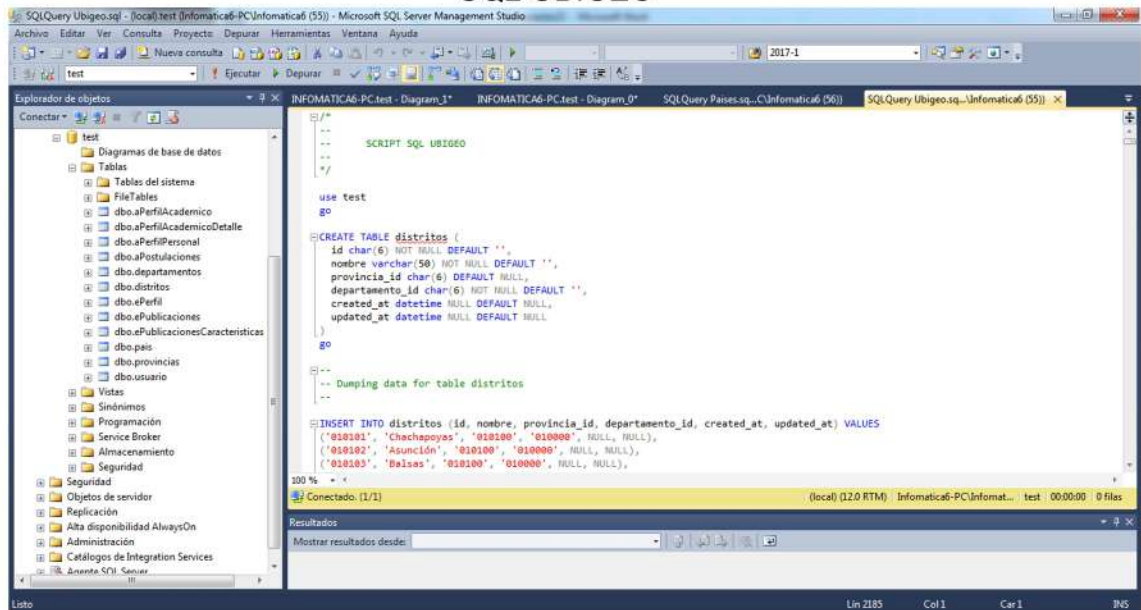
## SQL PAISES



The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The left pane displays the 'test' database structure, including tables like 'dbo.aPerfilAcademico' and 'dbo.país'. The main window shows a SQL query script for creating and populating the 'país' table.

```
/*  
--  
-- SCRIPT SQL PAISES  
--  
*/  
use test  
go  
  
CREATE TABLE país (  
id int PRIMARY KEY,  
iso char(2) DEFAULT NULL,  
nombre varchar(80) DEFAULT NULL  
)  
GO  
  
INSERT INTO país VALUES(1, 'AF', 'Afganistán');  
INSERT INTO país VALUES(2, 'AX', 'Islas Gland');  
INSERT INTO país VALUES(3, 'AL', 'Albania');  
INSERT INTO país VALUES(4, 'DE', 'Alemania');  
INSERT INTO país VALUES(5, 'AD', 'Andorra');  
INSERT INTO país VALUES(6, 'AO', 'Angola');  
INSERT INTO país VALUES(7, 'AI', 'Anguilla');  
INSERT INTO país VALUES(8, 'AQ', 'Antártida');  
INSERT INTO país VALUES(9, 'AG', 'Antigua y Barbuda');  
INSERT INTO país VALUES(10, 'AN', 'Antillas Holandesas');  
INSERT INTO país VALUES(11, 'SA', 'Arabia Saudí');  
INSERT INTO país VALUES(12, 'OE', 'Argelia');
```

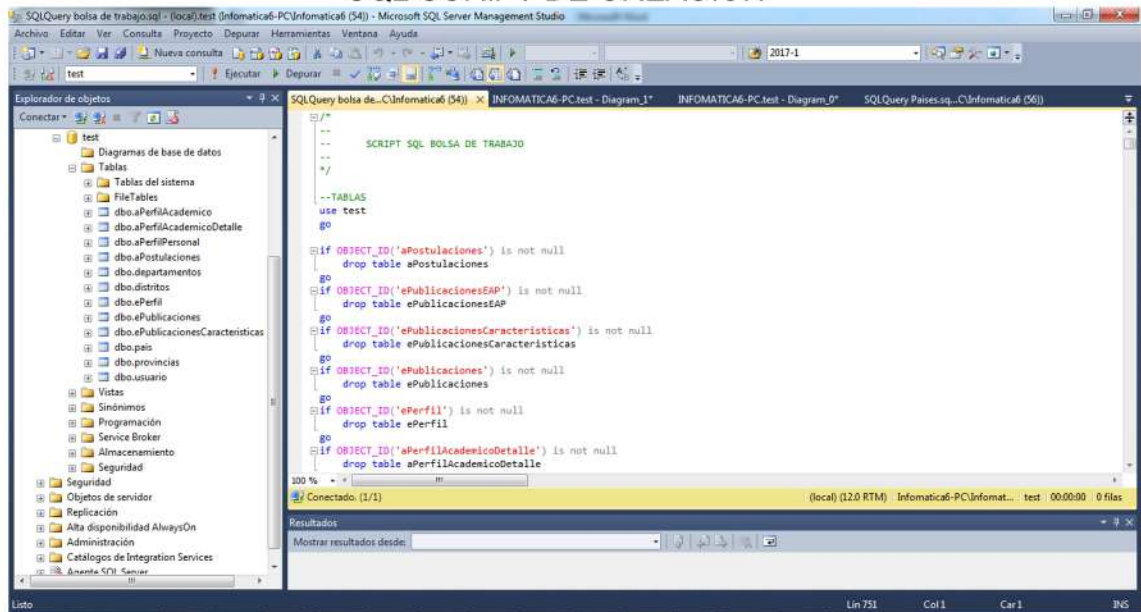
## SQL UBIGEO



The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The left pane displays the 'test' database structure. The main window shows a SQL query script for creating and populating the 'distritos' table.

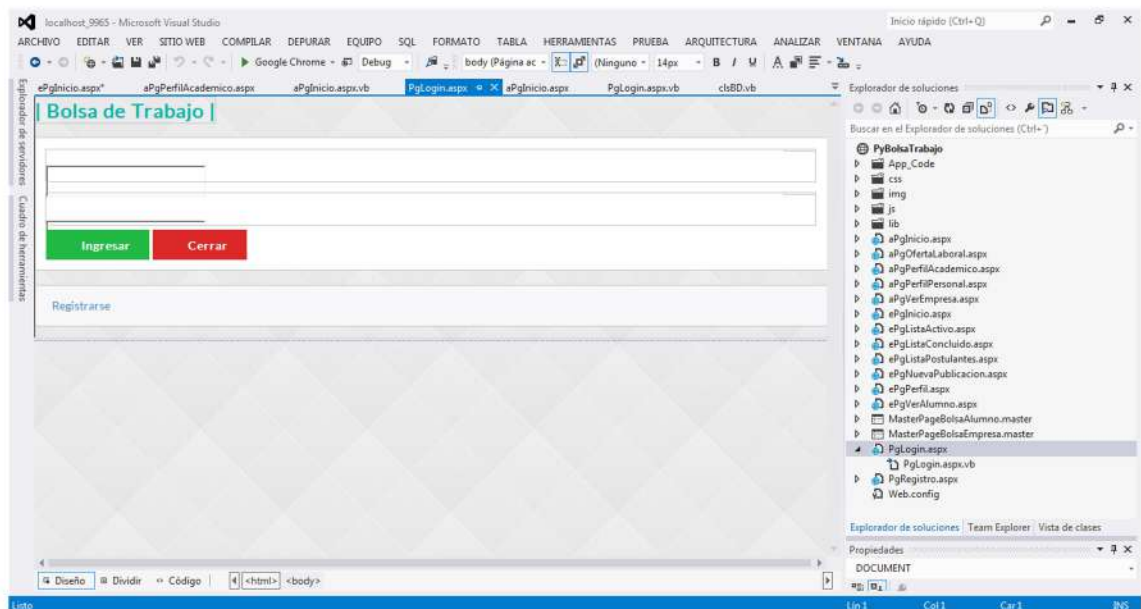
```
/*  
--  
-- SCRIPT SQL UBIGEO  
--  
*/  
use test  
go  
  
CREATE TABLE distritos (  
id char(6) NOT NULL DEFAULT '',  
nombre varchar(50) NOT NULL DEFAULT '',  
provincia_id char(6) DEFAULT NULL,  
departamento_id char(6) NOT NULL DEFAULT '',  
created_at datetime NULL DEFAULT NULL,  
updated_at datetime NULL DEFAULT NULL  
)  
GO  
  
-- Dumping data for table distritos  
  
INSERT INTO distritos (id, nombre, provincia_id, departamento_id, created_at, updated_at) VALUES  
(  
'010101', 'Chachapoyas', '010100', '010000', NULL, NULL),  
(  
'010102', 'Asunción', '010100', '010000', NULL, NULL),  
(  
'010103', 'Baizas', '010100', '010000', NULL, NULL),
```

## SQL SCRIPT DE CREACIÓN

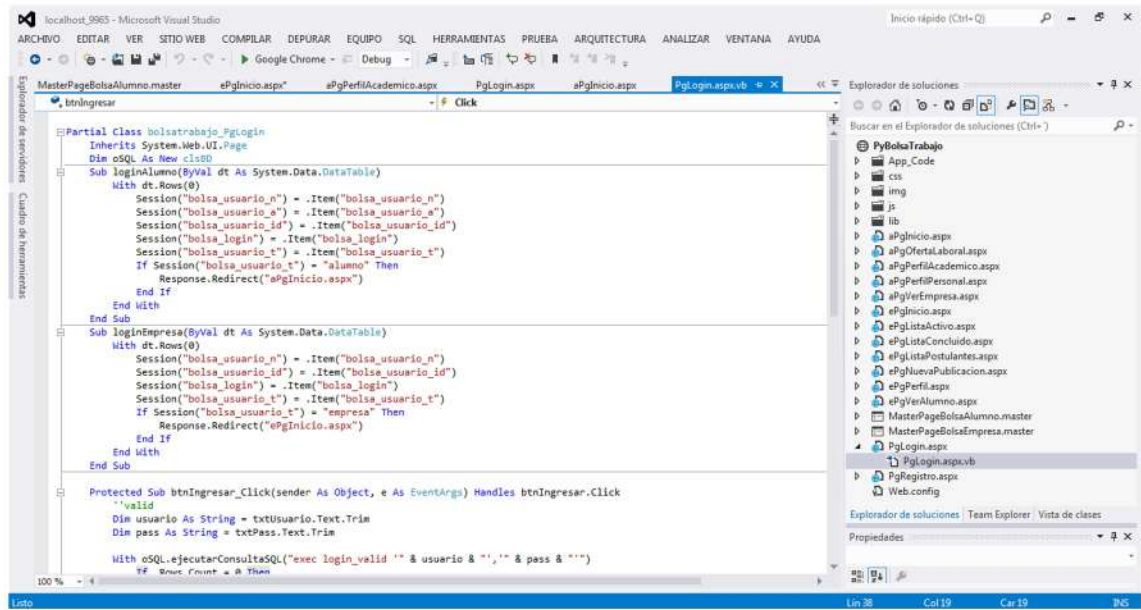


## EVIDENCIAS DE LA CREACIÓN DEL CÓDIGO FUENTE

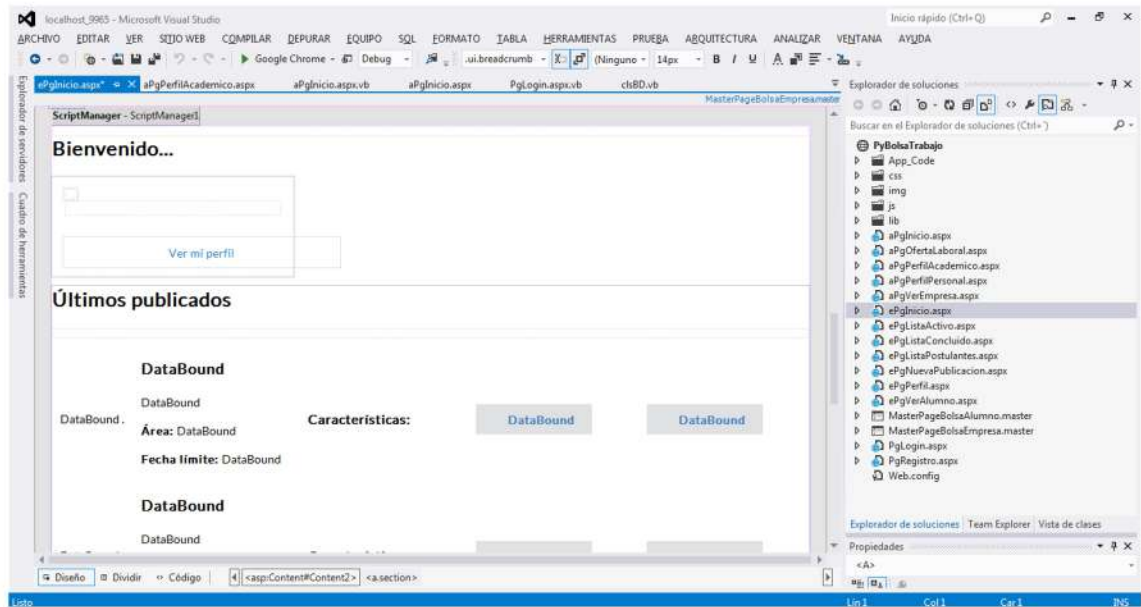
### VISUAL STUDIO – LOGIN



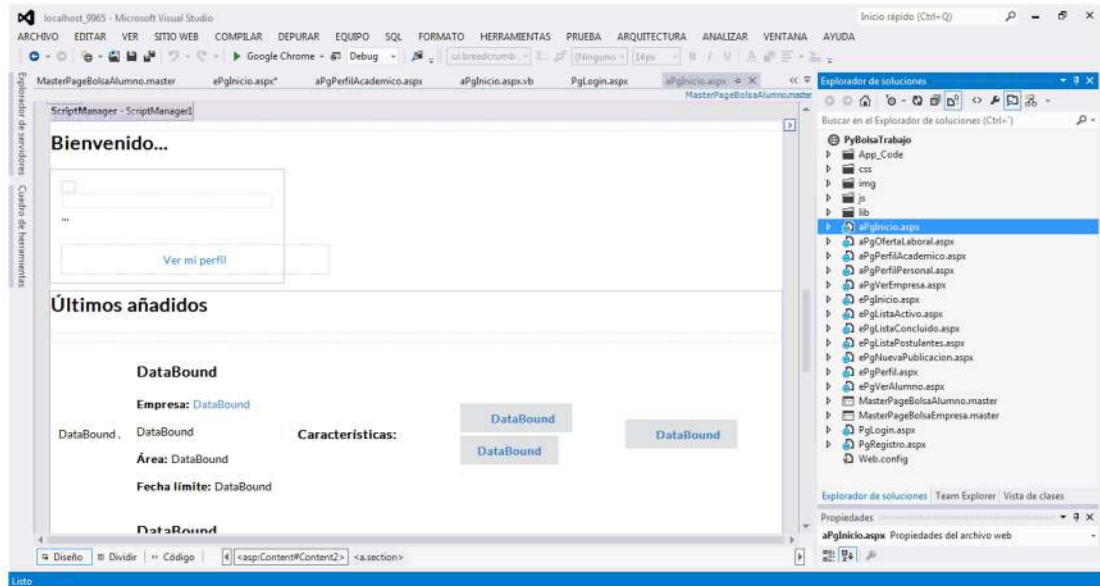
## VISUAL STUDIO – LOGIN CÓDIGO



## VISUAL STUDIO – PÁGINA INICIO EMPRESA



## VISUAL STUDIO – INICIO ALUMNO



OFICINA DE RED INFORMÁTICA - LUGAR DONDE SE REALIZÓ EL SISTEMA BOLSA DE TRABAJO DE LA UNIVERSIDAD DE HUÁNUCO.

