

UNIVERSIDAD NACIONAL TECNOLÓGICA DE LIMA SUR

FACULTAD DE INGENIERÍA Y GESTIÓN

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



**“PROTOTIPO DE BOLSA DE TRABAJO BASADO EN APLICACIONES
WEB PROGRESIVAS PARA LA MEJORA DEL PROCESO DE PRÁCTICAS
PRE PROFESIONALES Y SEGUIMIENTO AL EGRESADO EN LA
UNIVERSIDAD NACIONAL TECNOLÓGICA DE LIMA SUR”**

TRABAJO DE SUFICIENCIA PROFESIONAL

Para optar el Título Profesional de

INGENIERO DE SISTEMAS

PRESENTADO POR EL BACHILLER

LAURA LOPEZ, ALEXIS HONORATO

Villa El Salvador

2018

DEDICATORIA

A mi madre Angélica Lopez Cuestas por su esfuerzo constante, comprensión, y cariño sin límites.

A mi padre Mateo Laura Andia por su apoyo incondicional, sabios consejos y paciencia.

A mis hermanos y hermanas por hacer de mis días alegres y divertidos.

A mis tíos y tías por encaminarme a ser un hombre de bien, hecho y derecho. En especial a mi querida tía Diana Lopez Cuestas que desde el cielo sigue cuidándome.

A mis estimados amigos por estar conmigo en las buenas y en las malas.

AGRADECIMIENTO

A la Oficina Central de Calidad y Acreditación en especial al Doctor Frank Edmundo Escobedo Bailón por su constante asesoramiento para el desarrollo del presente trabajo de investigación.

A la Oficina de Tutoría y Seguimiento al Egresado en especial al Doctor Juan Carlos Oruna Lara por el apoyo y la información brindada.

A la Oficina General de Tecnologías de la Información y Comunicación en especial a la Ingeniera Denisse Marie Barrientos Pichilingue por la entrevista brindada, la cual fue de vital importancia para el desarrollo de la investigación.

A mis amigos y colegas Alexander Perez Mendoza, Gerard Chozo Navarte, Vladimir Peralta Navarte, Sheeffri Taype Zuñiga y Ernesto Espinoza Mejia por todo el conocimiento, experiencia y apoyo compartido, indispensable para la realización del presente trabajo de investigación.

ÍNDICE

INTRODUCCIÓN	10
CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA	13
1.1 Descripción de la Realidad Problemática.....	13
1.2 Justificación del Problema	15
1.3 Delimitación del Proyecto.....	18
1.3.1 Temporal.....	18
1.3.2 Espacial.....	18
1.4 Formulación del Problema	18
1.4.1 Problema General.	18
1.4.2 Problemas específicos.....	19
1.5 Objetivos	19
1.5.1 Objetivo General.....	19
1.5.2 Objetivos Específicos.....	19
CAPÍTULO II: MARCO TEÓRICO	20
2.1 Antecedentes	20
2.2 Bases Teóricas	25
2.2.1 Scrum.	25
2.2.2 Tipos de aplicaciones móviles según su desarrollo.....	57
2.2.3 Aplicaciones web progresivas	61
2.2.4 Manejo de control de versiones – GIT.	62
2.2.5 Ionic.	63
2.3 Definición de Términos Básicos.....	63
2.3.1 Aplicación web.	63
2.3.2 Arquitectura de una aplicación web.	64
2.3.3 Base de Datos.....	64

2.3.4	Bolsa de trabajo.....	64
2.3.5	Cliente / Servidor web.....	64
2.3.6	CSS3.....	65
2.3.7	Diseño web adaptable.....	65
2.3.8	Empresa.....	66
2.3.9	HTML5.....	66
2.3.10	Información.....	67
2.3.11	JavaScript.....	67
2.3.12	MySQL.....	68
2.3.13	Navegador web.....	68
2.3.14	Prototipo.....	68
2.3.15	Smartphone.....	69

CAPÍTULO III: DESARROLLO DEL TRABAJO DE SUFICIENCIA PROFESIONAL 70

3.1	Bolsa de trabajo.....	70
3.1.1	Objetivos.....	70
3.1.2	Alcance.....	71
3.1.3	Normas y referencias.....	71
3.1.4	Responsables.....	71
3.1.5	Líneas guía.....	71
3.2	Fase de Inicio.....	72
3.2.1	Creación de la visión del proyecto.....	72
3.2.2	Identificación del Scrum Master y los socios.....	81
3.2.3	Formación del equipo Scrum.....	83
3.2.4	Desarrollo de épicas.....	84
3.2.5	Creación de la lista priorizada de pendientes del producto.....	86

3.2.6	Realizar el plan de lanzamiento.....	89
3.3	Fase de Planificación y Estimación.....	91
3.3.1	Creación de historias de usuario.	91
3.3.2	Aprobación, estimación y asignación de historias de usuario.....	94
3.3.3	Creación de la lista de pendientes del Sprint.....	99
3.4	Fase de Implementación.....	100
3.4.1	Creación de entregables.	100
3.5	Fase de Revisión y Retrospectiva.	118
3.5.1	Demostración y validación del sprint.....	118
3.5.2	Herramienta.....	118
3.5.3	Retrospectiva de Sprint.	119
3.6	Fase de Lanzamiento.	120
3.6.1	Envío de entregables.	120
3.6.2	Retrospectiva del proyecto.	121
	CONCLUSIONES	123
	RECOMENDACIONES	124
	BIBLIOGRAFÍA	125
	ANEXOS	129
	ANEXO 1	129
	ANEXO 2	135

LISTADO DE FIGURAS

Figura 1.	Proceso SCRUM.....	57
Figura 2.	Mockup login	102
Figura 3.	Mockup mi perfil	102
Figura 4.	Mockup publicaciones.....	103
Figura 5.	Mockup detalle publicación	103
Figura 6.	Mockup notificaciones	104
Figura 7.	Mockup historial publicador	104
Figura 8.	Mockup nueva publicación	105
Figura 9.	Mockup editar publicación.....	105
Figura 10.	Mockup eliminar publicación	106
Figura 11.	Modelo de base de datos	107
Figura 12.	Creación del repositorio del proyecto	108
Figura 13.	Proyecto subido a Bitbucket.....	109
Figura 14.	Issues del proyecto en Bitbucket.....	110
Figura 15.	Maquetación login	111
Figura 16.	Maquetación mi perfil	111
Figura 17.	Maquetación publicaciones	112
Figura 18.	Maquetación detalle publicación	112
Figura 19.	Maquetación notificaciones	113
Figura 20.	Maquetación historial publicador	114
Figura 21.	Maquetación nueva publicación	114
Figura 22.	Maquetación editar publicación.....	115
Figura 23	Maquetación eliminar publicación	115

Figura 24. Commits del proyecto en Bitbucket.....	116
Figura 25. Conexión de base de datos.....	116
Figura 26. Inicio de sesión	117
Figura 27. Modulo Login.....	117
Figura 28. Valoración de la bolsa de trabajo según desempeño.....	132
Figura 29. Problemas encontrados vs estudiantes.....	132
Figura 30. Porcentaje de estudiantes que usan Smartphone u otro dispositivo móvil.....	133
Figura 31. Estudiantes que utilizarían la aplicación.....	134
Figura 32. Funcionalidades vs estudiantes	134
Figura 33. Login estudiante	135
Figura 34. Menú principal.....	136
Figura 35. Publicaciones	136
Figura 36. Detalle de publicación	137
Figura 37. Postular	137
Figura 38. Notificaciones.....	138
Figura 39. Mi perfil.....	139
Figura 40. Login administrador.....	140
Figura 41. Menú principal.....	141
Figura 42. Historial publicaciones.....	142
Figura 43. Nueva publicación	142
Figura 44. Editar y eliminar publicación.....	143
Figura 45. Administrador	143

LISTADO DE TABLAS

Tabla 1	Ejemplo de Historia de Usuario	47
Tabla 2	Ejemplo de Product Backlog	50
Tabla 3	Ejemplo de Sprint Backlog	53
Tabla 4	Riesgos identificados	74
Tabla 5	Coste de recursos humanos	80
Tabla 6	Coste de recursos hardware y software	81
Tabla 7	Product Backlog del proyecto.....	87
Tabla 8	Cronograma de planificación del lanzamiento.....	90
Tabla 9	Historia de usuario 1	95
Tabla 10	Historia de usuario 2	96
Tabla 11	Historia de usuario 3	96
Tabla 12	Historia de usuario 4	96
Tabla 13	Historia de usuario 5	97
Tabla 14	Historia de usuario 6	97
Tabla 15	Historia de usuario 7	97
Tabla 16	Historia de usuario 8	98
Tabla 17	Historia de usuario 9	98
Tabla 18	Historia de usuario 10	98
Tabla 19	Especificación de los Sprint Backlog.....	100

INTRODUCCIÓN

El presente trabajo de investigación lleva por título **“PROTOTIPO DE BOLSA DE TRABAJO BASADO EN APLICACIONES WEB PROGRESIVAS PARA LA MEJORA DEL PROCESO DE PRÁCTICAS PRE PROFESIONALES Y SEGUIMIENTO AL EGRESADO EN LA UNIVERSIDAD NACIONAL TECNOLÓGICA DE LIMA SUR”**, para optar por el título de “Ingeniero de Sistemas”, presentado por el bachiller Laura Lopez, Alexis Honorato.

La Universidad Nacional Tecnológica de Lima Sur (UNTELS) es una universidad relativamente nueva, que viene formando profesionales altamente calificados, con valores, vocación de servicio y compromiso con la sociedad. Uno de sus objetivos para con sus estudiantes es brindarles las herramientas necesarias para lograr su inserción en el mercado laboral, para ello contaba con la Oficina de Tutoría y Seguimiento al Egresado que tenía entre algunas de sus funciones la gestión de prácticas pre profesionales, la administración de la bolsa de trabajo, las tutorías y el seguimiento a los egresados. Si bien es cierto que cumplía con las funciones para las que fue creada, demostró tener graves deficiencias en los procesos de publicación y difusión de convocatorias laborales, así como una inadecuada comunicación con los estudiantes y egresados. Actualmente, estas funciones han pasado a formar parte de la responsabilidad directa de las Oficinas de las Carreras Profesionales, y pensando en ello es necesario automatizar dichos procesos para ofrecer una atención más efectiva hacia los estudiantes.

Ante este problema el presente trabajo busca mostrar los beneficios que traería el desarrollo de un prototipo de bolsa de trabajo basado en aplicaciones web

progresivas. Para el desarrollo del proyecto se usó la metodología Scrum, pues es una de las mejores opciones para desarrollar del presente proyecto, permite gestionar las expectativas del cliente en base a resultados tangibles, proporciona flexibilidad a los cambios, mitigando riesgos al tener como base a un equipo de trabajo auto gestionado y comprometido.

Cabe señalar que gracias a las características que posee una aplicación web progresiva, se logró desarrollar un prototipo de bolsa de trabajo, el cual permitió automatizar los procesos, para la gestión de prácticas pre profesionales y seguimiento al egresado. Con esto se redujo el tiempo empleado de cinco minutos a un minuto.

El análisis mediante una encuesta online, realizado a los alumnos de la UNTELS, permitió identificar los problemas de la bolsa de trabajo. Scrum a través de sus herramientas como la reunión de la visión del proyecto, en la que participaron el propietario del producto y los socios, se logró identificar los requerimientos para el desarrollo del proyecto y las expectativas de los socios. Así mismo por medio de las historias de usuario se logró documentar de forma simple, los requerimientos y funcionalidades que el usuario final deseaba.

El uso de la metodología Scrum como marco de trabajo facilitó que los requerimientos sean atendidos de manera rápida y ajustable, gracias a la constante comunicación entre todas las partes del proyecto. Así mismo permitió gestionar el desarrollo del proyecto con un enfoque iterativo e incremental, siguiendo las fases definidas, con sus procesos bien estructurados. Los roles, eventos y artefactos de Scrum al ser bien detallados y fáciles de usar permitieron maximizar el valor del producto, a la vez que se redujeron los recursos empleados.

Las aplicaciones web progresivas al combinar lo mejor de la web tradicional y las aplicaciones móviles nativas, permitieron que el prototipo funcione para todos los usuarios, sin importar el dispositivo, sistema operativo o navegador usado. Gracias al Shell app se logró navegación, interacción y apariencia muy similar a las aplicaciones móviles nativas. También se logró navegar con redes débiles o sin conexión, recibir notificaciones push y mantenerse actualizada, gracias al Service Worker.

El presente trabajo se ha desarrollado en 3 capítulos y consta de los siguientes capítulos: Capítulo 1 se plantea el problema y se enuncian los objetivos del trabajo, en el capítulo 2 se desarrolla el marco teórico, antecedentes, bases teóricas, definición de términos básicos y en el capítulo 3 se describe el desarrollo del trabajo de suficiencia profesional.

CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA

1.1 Descripción de la Realidad Problemática

La Universidad Nacional Tecnológica de Lima Sur (UNTELS) es una casa de estudios relativamente nueva, el 10 de enero de 2001 fue creada la Universidad Nacional Tecnológica del Cono Sur, mediante la Ley N° 27431 promulgada por el Gobierno de Transición Democrática presidido por Valentín Paniagua Corazao. El 29 de setiembre de 2005, por disposición del Ministro de Educación, Javier Sota Nadal, se constituyó la primera Comisión Organizadora y el 15 de marzo del 2006 se entregó a la universidad el terreno de su Sede Central, así como el presupuesto destinado para la construcción del primer pabellón de aulas en el campus universitario. El 6 de noviembre de ese año se aprobó su Proyecto de Desarrollo Institucional. El 25 de febrero del año 2007 se efectuó el primer concurso público de admisión y el 1 de abril de ese mismo año se dio inicio al primer año académico.

Mediante la Ley N.º 30184, de fecha 6 de mayo de 2014, pasó a denominarse Universidad Nacional Tecnológica de Lima Sur. Actualmente, ofrece 5 carreras profesionales y tiene un aproximado de tres mil estudiantes.

Esta universidad está en proceso de obtener el licenciamiento institucional, para lo cual debe cumplir con las Condiciones Básicas de Calidad necesarias para ofrecer el servicio educativo superior.

Según la Oficina Central de Calidad y Acreditación, entre los años 2012 y 2016 egresaron un total de 800 estudiantes, de los cuales no se tiene ninguna información sobre su situación laboral actual, pues la UNTELS hasta antes de mediados del 2017 no contaba con la función de seguimiento al egresado en el documento normativo Reglamento de Organización y Funciones, el cual debía estar aprobado por la autoridad competente, donde se especifique las funciones del área, dirección o departamento encargado del seguimiento del egresado. Esta situación casi no ha mejorado, ya que a pesar de que existe formalmente la función de seguimiento al egresado, solo se lleva control de la cantidad de egresados y titulados. En lo que se refiere al seguimiento del egresado por el momento están elaborando una encuesta para conocer la situación laboral del egresado.

Hasta fines de enero del presente año, la gestión de las prácticas pre profesionales, el seguimiento al egresado, la ejecución de asesorías académicas y la administración de la bolsa de trabajo estaba a cargo de la Oficina de Tutoría y Seguimiento al Egresado que, si bien es cierto, cumplía con sus funciones, también tenía graves problemas. Posteriormente esta oficina fue cerrada y sus funciones pasaron a manos de cada oficina de carrera profesional.

La bolsa de trabajo de la UNTELS permite a sus alumnos postular a convocatorias laborales, está estrechamente relacionada con la gestión de prácticas pre profesionales y el seguimiento al egresado. Esta bolsa de trabajo

viene arrastrando problemas que le impide brindar un servicio adecuado, el principal problema es que sus procesos no están automatizados. Así mismo la publicación de convocatorias laborales aún se hace a través de medios impresos, una práctica poco amigable con el medio ambiente y que limita la cantidad de estudiantes a los que puede llegar, la sección bolsa de trabajo no se encuentra en lugar fácil de ubicar, dentro de la página web de la UNTELS, también presenta fallas funcionales y de diseño. Los procesos de registro y postulación siguen siendo de modo presencial, con documentos impresos, no se ha implementado una forma de notificar cada vez que se realiza una nueva publicación, no funciona sin conexión a internet o con redes de mala calidad, carece de una forma rápida y fácil de acceder a la bolsa de trabajo.

Ante estos problemas, el presente proyecto plantea el desarrollo de un prototipo de bolsa de trabajo basado en aplicaciones web progresivas que permita automatizar el proceso de prácticas pre profesionales y seguimiento al egresado. Se optó por esta alternativa debido a las ventajas que una aplicación web progresiva ofrece, pues reúne las mejores características tanto de las aplicaciones móviles nativas como de las aplicaciones web tradicionales.

1.2 Justificación del Problema

El 2 marzo del año 2018, se realizó una encuesta online a los alumnos de la UNTELS, acerca de la bolsa de trabajo de su universidad. Contó con un total de 378 participantes de los cuáles el 52% de los encuestados consideran que la bolsa de trabajo tiene un desempeño regular, mientras que el 46% consideran que es malo y solo 2% considera que es buena. Así mismo los problemas más frecuentes identificados fueron “no sabe cuándo se realiza una

nueva publicación o actualización”, “no puede hallar fácilmente la bolsa de trabajo en la página web de la UNTELS”, “no me notifican a tiempo si logré ser uno de los seleccionados”. Esta encuesta se encuentra en el anexo 1.

Estos resultados demuestran que existen problemas en la bolsa de trabajo una de las causas es que los procesos no están automatizados esto se ve reflejado en la baja cantidad de estudiantes a los que llega, la dificultad para acceder a la bolsa de trabajo, la manera poco eficiente como se realiza la publicación de convocatorias laborales y el casi nulo seguimiento que se le hace al practicante o egresado.

Los Mecanismos de inserción laboral, como las bolsas de trabajo universitarias son una de las Condiciones Básicas de Calidad, necesarias para ofrecer el servicio educativo superior, el cual es un requisito para que las universidades puedan acceder al Licenciamiento Institucional, una razón más para contar con una bolsa de trabajo que cumpla con los estándares de calidad.

Se eligió las aplicaciones web progresivas en lugar de otras opciones, debido a los beneficios que estas brindan. A continuación, se describen las principales características de esta tecnología.

Multiplataforma pues funcionan en diversos sistemas operativos, con estilo app, rápida y de fácil navegación. Adaptable a cualquier dispositivo, sea de escritorio, móvil o tablet, además de vinculable, se puede compartir fácilmente vía URL.

Descubrible, permite que los motores de búsqueda la encuentren, con la posibilidad de volver a interactuar con el usuario, a través de funciones como las notificaciones push. Funcionan independientemente de la conectividad, pues puede trabajar sin conexión o con redes de mala calidad.

Instalable, permite conservar las apps en la pantalla principal sin la molestia de una tienda de app, con un costo y tiempo de desarrollo razonable a comparación de aplicaciones nativas o híbridas.

A continuación, se describen las principales razones que justifican el desarrollo del presente proyecto.

Los estudiantes contarán con un usuario y contraseña lo que impedirá suplantación de identidad u otros problemas de seguridad. Luego de iniciar sesión en el prototipo, los usuarios podrán acceder a las últimas publicaciones, información relevante y la función de postular a una convocatoria laboral vigente.

La recolección de requisitos para postular a una convocatoria laboral será de forma virtual, es decir se enviarán por medio de la bolsa de trabajo, con esto se reducirá el uso de tinta y papel. Se automatizarán los procesos de registros y postulaciones, con esto se reducirá el tiempo en los trámites y también se reducirán los errores cometidos por el personal administrativo.

A través de notificaciones, se alertará a los usuarios de las nuevas publicaciones, lo cual permitirá llegar a más usuarios.

Para el desarrollo del proyecto se usará la metodología Scrum, esto permitirá que el desarrollo sea ágil, se adapte a los cambios o problemas que pueden ir surgiendo. También permitirá que el equipo de desarrollo sea auto gestionado y esté comprometido.

Con el desarrollo de este proyecto también se busca que el prototipo desarrollado sirva como base para futuras investigaciones que permitan desarrollar aplicaciones web progresivas que abarquen más procesos de la gestión universitaria.

1.3 Delimitación del Proyecto

1.3.1 Temporal.

El presente proyecto comprende el período de marzo hasta junio del año 2018.

1.3.2 Espacial.

El presente proyecto se realizará en la Universidad Nacional Tecnológica de Lima Sur (UNTELS).

El prototipo funcionará como un aplicativo de bolsa de trabajo basada en aplicaciones web progresivas, para los estudiantes y egresados de la UNTELS. Al ser un aplicativo web progresivo ofrecerá una experiencia de usuario similar a una aplicación móvil nativa, también se verá beneficiada de las ventajas multiplataforma, al poder funcionar en Android, iOS, Windows, macOS, Linux entre otros. Así mismo será compatible con navegadores como Chrome, Safari, Mozilla Firefox, entre otros.

Al ser un prototipo con fines demostrativos, será limitada, sirviendo de modelo o base a futuras aplicaciones.

1.4 Formulación del Problema

1.4.1 Problema General.

¿Cómo desarrollar un prototipo de bolsa de trabajo basada en aplicaciones web progresivas que permita automatizar los procesos de prácticas pre profesionales y seguimiento al egresado en la Universidad Nacional Tecnológica de Lima Sur?

1.4.2 Problemas específicos.

¿Cómo identificar los requerimientos para el desarrollo de un prototipo de bolsa de trabajo?

¿Cómo establecer Scrum como el marco de trabajo para el desarrollo de un prototipo de bolsa de trabajo?

¿Cómo el uso de las características de aplicaciones web progresivas permitirá dotar de las funcionalidades requeridas al prototipo de bolsa de trabajo?

1.5 Objetivos

1.5.1 Objetivo General.

Desarrollar un prototipo de bolsa de trabajo basado en aplicaciones web progresivas para automatizar los procesos de prácticas pre profesionales y seguimiento al egresado en la Universidad Nacional Tecnológica de Lima Sur.

1.5.2 Objetivos Específicos.

Identificar los requerimientos para el desarrollo de un prototipo de bolsa de trabajo.

Establecer Scrum como el marco de trabajo para el desarrollo del prototipo de bolsa de trabajo.

Usar las características de las aplicaciones web progresivas para dotar de las funcionalidades requeridas al prototipo de bolsa de trabajo.

CAPÍTULO II: MARCO TEÓRICO

2.1 Antecedentes

Espinoza, L, y Pérez, J. (2009). Sistema Web de “Bolsa de Empleo” para el Ministerio de Trabajo y Empleo del Ecuador (Tesis de pregrado). Universidad de las Fuerzas Armadas ESPE, Sangolquí, Ecuador. Busca proporcionar una herramienta informática que permita minimizar el índice de desempleo dentro de Ecuador mediante un sistema web, que optimice el uso de recursos y satisfacer a los usuarios. Se logró automatizar los procesos dentro del Servicio Público de Empleo, como resultado se minimizaron el uso de recursos, como la reducción del tiempo, también se redujo gastos mediante el uso de software libre. Las interfaces utilizadas son de fácil uso y proporcionan una mejor experiencia de usuario.

Gómez, G. (2010). Sistema de información para bolsa de trabajo (Tesis de pregrado). Universidad Veracruzana, Veracruz, México. Tiene por objetivo desarrollar un sistema de información que permita a los egresados de los tres programas de estudio que imparte la Facultad de Contaduría y Administración, vincularse con el mercado laboral. Se llegó a la conclusión que la búsqueda de

trabajo sin salir de casa y sin gastar dinero en imprimir el curriculum, es una oportunidad que nos brinda internet a la vez que es una herramienta importante en la captación y selección de personal, permitiendo la reducción de recursos que antiguamente se destinaba para este proceso. La implementación de una bolsa de trabajo logrará la vinculación de la facultad con los sectores productivos público, privado y con diversas instituciones, que le permitirá fortalecer vínculos y proporcionar a sus egresados y estudiantes la oportunidad de contar con un empleo de acuerdo con su perfil profesional.

Quispe, R. (2012). Bolsa de trabajo carrera de informática (Tesis de pregrado). Universidad Mayor de San Andrés, La Paz, Bolivia. Tiene por objetivo publicar las convocatorias laborales y promocionar a los egresados de la carrera, solucionando así la falta de una herramienta tecnológica que ayude a gestionar los requerimientos de empleo, de las instituciones y de los egresados de la carrera. Se llegó a concluir que, bajo las especificaciones planteadas, y siguiendo la metodología Programación Extrema para el desarrollo, todo fue de acuerdo a lo planeado, se realizó una interfaz amigable, de fácil acceso a los diferentes módulos que contiene el sistema, así mismo se construyó una base de datos, que se integró al Sistema Académico de la carrera de Informática (SIA), para trabajar de manera conjunta.

Torres, M. (2011). Propuesta de una aplicación (Web) móvil en la Facultad de Contaduría, Administración y Sistemas (Tesis de pregrado). Universidad Veracruzana, Veracruz, México. Plantea el problema de la página web de la Universidad Veracruzana, la cual, a pesar de contener información muy valiosa, no estaba optimizada para ser accedida desde dispositivos móviles. Por ello la autora propone el desarrollo de una aplicación para el

sistema operativo Windows Phone 7 que utilizará el navegador móvil, ésta mostrará los eventos próximos a realizarse en su Universidad y otras funciones accesibles a través de usuario y contraseña como son la revisión de horarios, notas del período actual, entre otros. El trabajo llega a la conclusión que dependiendo de las necesidades es válido optar por una aplicación móvil como por una aplicación web, se debe decidir a qué sistema operativo móvil orientarse, aunque esto es un limitante del público al que se puede llegar. Así mismo se debe asegurar la compatibilidad con los diferentes navegadores web, finalmente remarca que si una aplicación pretende llegar a un público consumidor masivo se debe tener en cuenta la accesibilidad a los dispositivos móviles.

Sauceda, G. (2008). Diseño de una bolsa de trabajo en el área de recursos humanos de una universidad (Tesis de maestría). Instituto Tecnológico de Sonora, Sonora, México. Tenía por objetivo diseñar una bolsa de trabajo en el área de recursos humanos de una universidad para estudiantes y egresados con la finalidad de agilizar su inserción con el sector empresarial en base a su perfil. Se concluyó que la metodología implementada, para el diseño de una bolsa de trabajo, proporciona una perspectiva de seguimiento más clara ya que se puede analizar las fases o elementos que se deben contemplar para un buen manejo y administración de la misma, así mismo se resaltó la importancia de contar con esta información documentada para la operación del responsable de recursos humanos, de la Universidad para la cual se realizó este diseño. El análisis realizado sobre los distintos modelos de bolsa de trabajo, permitió compararlos para optar por el que mejor se acomode a las características y necesidades del área para la cual se diseñó.

Alania, J, y Apaza, N. (2014). Aplicación web de bolsa de empleo de empresas y personas utilizando el API de Google Maps para la accesibilidad a la información de oportunidades laborales en la región Puno, 2013 (Tesis de pregrado). Universidad Nacional del Altiplano, Puno, Perú. Tiene por objetivo desarrollar la aplicación web de Bolsa de Empleo de empresas y personas utilizando la API de Google Maps para ayudar a la accesibilidad a la información de oportunidades laborales en la región Puno. Se llegó a la conclusión que la aplicación web de bolsa de empleo, utilizando el API de Google Maps, PHP para la lógica de negocio, MySQL como gestor de la base de datos y las tecnologías Ajax y jquery para el desarrollo del sistema, ayudo a obtener información geográfica de las empresas, además de proporcionar información laboral del forma rápida e intuitiva. Se realizó una comparación entre agosto 2013 y octubre 2013 sin aplicación web y con aplicación web respectivamente, obteniéndose en el periodo agosto 2013 una calificación de 2.44 puntos mientras que en el periodo octubre 2013 se obtuvo una calificación de 4.59 puntos los cuales fueron dados por personas desempleadas. Aplicando la prueba t – student con una significancia de 5% se obtuvo la $t_{calculada}=15.36 > t_{tabulada}=4.30$ concluyendo que la aplicación web de bolsa de empleo de empresas y personas utilizando el API de Google Maps ayuda a la accesibilidad a la información de oportunidades laborales en la región Puno.

Yaya, B. (2015). Sistema de información para bolsa de trabajo en el Instituto de Educación Superior Tecnológico Público Argentina 2015 (Tesis de maestría). Universidad Nacional Hermilio Valdizán, Lima, Perú. Tiene por objetivo desarrollar el sistema de información para la bolsa de trabajo del Instituto de Educación Superior Tecnológico Público Argentina. Se concluyó en

base al estudio hecho que el 73.6% ha llevado o llevaría su hoja de vida a una bolsa de empleo, mientras que el 26.4 no lo haría. El 31.1% trabaja o a trabajado por intermedio de una bolsa de empleo y el 68.9% estante no. Un 62.9% tiene una imagen regular de las bolsas de empleo, 32.1% buena y 5.1% mala.

Raffo, E., Ruiz, E., Lorena, P., Tupia, E., Pérez, V. y Mayta R. (2002). Implementación de la bolsa de trabajo on line en la facultad de ingeniería industrial – UNMSM. *Industrial Data*, 5(2), 44-51. Detalla el análisis, diseño e implementación de la Bolsa de Trabajo On Line, como parte del PORTAL INDUSTRIAL UNMSM. Surge ante la necesidad de buscar una forma más eficiente de gestionar y administrar los tramites de la Oficina de Prácticas Profesionales. Se llegó a la conclusión que el portal de uso exclusivo para estudiantes y egresados, con un control de seguridad, actualización de la base de datos de la bolsa de trabajo y mantenimiento remoto es muy beneficioso debido a lo antes mencionado como también por los mínimos costos de mantenimiento, con la posibilidad de ser comercializado.

Aguirre, L., y Sinche H. (2013). Diseño de una aplicación móvil para la consulta académica de la FIIS-UTP (Tesis de pregrado). Universidad Tecnológica del Perú, Lima, Perú. Tiene por objetivo diseñar una móvil para la consulta académica de la Facultad de Ingeniería Industrial y de Sistemas de la Universidad Tecnológica del Perú, aprovechando el avance de los teléfonos inteligentes, este aplicativo móvil se encargará de las consultas académicas más demandadas del portal web de la Universidad Tecnológica del Perú, al cual según explican los autores se accede mayoritariamente a través de un ordenador. El aplicativo móvil facilitó a los estudiantes acceder a su información

académica, como por ejemplo averiguación de notas después de un periodo prácticas.

Universidad de Lima (UL, 2014). La Universidad de Lima y su aplicación ULIMA APP que está disponible para los sistemas operativos Android y iOS, disponibles en Google Play y App Store respectivamente, la cual se encarga de mostrar noticias, conectar con el aula virtual en la que estudiantes pueden revisar sus calificaciones, mostrar el menú de la cafetería, informar sobre números telefónicos de atención, realizar pagos de mensualidades, ver horarios, etc. Esta aplicación fue desarrollada por un equipo de 7 personas pertenecientes al IT LAB de la carrera de Ingeniería de Sistemas de la universidad.

2.2 Bases Teóricas

2.2.1 Scrum.

Scrum es una metodología ágil utilizada para gestionar el desarrollo de proyectos; no es una técnica que describa como realizar un producto, más bien es un marco de trabajo en que pueden emplearse distintos procesos y técnicas. Scrum nos enseña la eficacia relativa entre las prácticas de gestión de producto y prácticas de desarrollo de manera que podemos mejorar (Schwaber & Sutherland, 2013).

2.2.1.1 Visión general de Scrum.

Schwaber & Sutherland (2013) Afirman que Scrum es un marco

de trabajo por el cual las personas pueden abordar problemas complejos, a la vez que entregan productos del máximo valor posible productiva y creativamente. Scrum es:

- Ligerero.
- Fácil de entender.
- Extremadamente difícil de llegar a dominar.

2.2.1.2 Teoría de Scrum.

Scrum está basada en la teoría de control de procesos empírica o empirismo. El empirismo afirma que el conocimiento procede de la experiencia y de tomar decisiones basándose en lo que se conoce. Scrum emplea un enfoque iterativo e incremental para optimizar la predictibilidad y el control del riesgo (Schwaber & Sutherland, 2013).

Tres pilares soportan toda la implementación del control de procesos empírico: transparencia, inspección y adaptación.

2.2.1.2.1 *Transparencia.*

Los aspectos significativos del proceso deben ser visibles para aquellos que son responsables del resultado. La transparencia requiere que dichos aspectos sean definidos por un estándar común, de tal modo que los observadores compartan un entendimiento común de lo que se está viendo. Aquellos que desempeñan el trabajo y aquellos que aceptan el producto de dicho trabajo deben compartir una definición común de “Terminado”. (Schwaber & Sutherland, 2013, p.5)

2.2.1.2.2 *Inspección.*

Los usuarios de Scrum deben inspeccionar frecuentemente los artefactos de Scrum y el progreso hacia un objetivo, para detectar variaciones. Su inspección no debe ser tan frecuente como para que interfiera en el trabajo. Las inspecciones son más beneficiosas cuando se realizan de forma diligente por inspectores expertos, en el mismo lugar de trabajo. (Schwaber & Sutherland, 2013, p.5)

2.2.1.2.3 *Adaptación.*

Si un inspector determina que uno o más aspectos de un proceso se desvían de límites aceptables, y que el producto resultante no será aceptable, el proceso o el material que está siendo procesado deben ser ajustados. Dicho ajuste debe realizarse cuanto antes para minimizar desviaciones mayores. (Schwaber & Sutherland, 2013, p.5)

2.2.1.2.4 *Manifiesto Ágil*

Son los valores sobre los que se asientan los métodos ágiles (Menzinsky, López, & Palacio, 2016).

- A los individuos y su interacción, por encima de los procesos y las herramientas.
- El software que funciona, por encima de la documentación exhaustiva.
- La colaboración con el cliente, por encima de la negociación contractual.

- La respuesta al cambio, por encima del seguimiento de un plan.

Scrum está compuesto por el Equipo de trabajo, los roles que desempeñan, los eventos, los artefactos realizados y las reglas asociadas. Cada componente tiene un fin específico que debe cumplirse para el éxito de Scrum (Caudevilla, 2010).

2.2.1.3 Roles de SCRUM.

2.2.1.3.1 El Dueño de Producto (Product Owner).

El Dueño de Producto es el responsable de maximizar el valor del producto y del trabajo del Equipo de Desarrollo. El cómo se lleva a cabo esto podría variar ampliamente entre distintas organizaciones, Equipos Scrum e individuos. El Dueño de Producto es la única persona responsable de gestionar la Lista del Producto (Product Backlog). (Schwaber & Sutherland, 2013, p.6)

La gestión de la Lista del Producto incluye:

- Expresar claramente los elementos de la Lista del Producto.
- Ordenar los elementos en la Lista del Producto para alcanzar los objetivos y misiones de la mejor manera posible.
- Optimizar el valor del trabajo desempeñado por el Equipo de Desarrollo.
- Asegurar que la Lista del Producto es visible, transparente y clara para todos, y que muestra aquello en lo que el equipo trabajará a continuación.

- Asegurar que el Equipo de Desarrollo entiende los elementos de la Lista del Producto al nivel necesario.

El Dueño de Producto podría hacer el trabajo anterior, o delegarlo en el Equipo de Desarrollo. Sin embargo, en ambos casos el Dueño de Producto sigue siendo el responsable de dicho trabajo.

El Dueño de Producto es una única persona, no un comité. El Dueño de Producto podría representar los deseos de un comité en la Lista del Producto, pero aquellos que quieran cambiar la prioridad de un elemento de la Lista deben hacerlo a través del Dueño de Producto.

Para que el Dueño de Producto pueda hacer bien su trabajo, toda la organización debe respetar sus decisiones. Las decisiones del Dueño de Producto se reflejan en el contenido y en la priorización de la Lista del Producto. No está permitido que nadie pida al Equipo de Desarrollo que trabaje con base en un conjunto diferente de requerimientos, y el Equipo de Desarrollo no debe actuar con base en lo que diga cualquier otra persona. (Schwaber & Sutherland, 2013, p.6)

2.2.1.3.2 *El Equipo de Desarrollo (Development Team).*

El Equipo de Desarrollo consiste en los profesionales que desempeñan el trabajo de entregar un Incremento de producto “Terminado”, que potencialmente se pueda poner en producción, al final de cada Sprint. Solo los miembros del Equipo de Desarrollo participan en la creación del Incremento.

Los Equipos de Desarrollo son estructurados y empoderados por la organización para organizar y gestionar su propio trabajo. La sinergia resultante optimiza la eficiencia y efectividad del Equipo de Desarrollo. (Schwaber & Sutherland, 2013, p.7)

Los Equipos de Desarrollo tienen las siguientes

características:

- Son autoorganizados. Nadie (ni siquiera el Scrum Master) indica al Equipo de Desarrollo cómo convertir elementos de la Lista del Producto en Incrementos de funcionalidad potencialmente desplegados.
- Los Equipos de Desarrollo son multi funcionales, contando como equipo con todas las habilidades necesarias para crear un Incremento de producto.
- Scrum no reconoce títulos para los miembros de un Equipo de Desarrollo, todos son Desarrolladores, independientemente del trabajo que realice cada persona; no hay excepciones a esta regla.
- Scrum no reconoce sub-equipos en los equipos de desarrollo, no importan los dominios particulares que requieran ser tenidos en cuenta, como pruebas o análisis de negocio; no hay excepciones a esta regla.
- Los Miembros individuales del Equipo de Desarrollo pueden tener habilidades especializadas y áreas en las que estén más

enfocados, pero la responsabilidad recae en el Equipo de Desarrollo como un todo.

Tamaño del Equipo de Desarrollo

El tamaño óptimo del Equipo de Desarrollo es lo suficientemente pequeño como para permanecer ágil y lo suficientemente grande como para completar una cantidad de trabajo significativa. Tener menos de tres miembros en el Equipo de Desarrollo reduce la interacción y resulta en ganancias de productividad más pequeñas. Los Equipos de Desarrollo más pequeños podrían encontrar limitaciones en cuanto a las habilidades necesarias durante un Sprint, haciendo que el Equipo de Desarrollo no pudiese entregar un Incremento que potencialmente se pueda poner en producción. Tener más de nueve miembros en el equipo requiere demasiada coordinación. Los Equipos de Desarrollo grandes generan demasiada complejidad como para que pueda gestionarse mediante un proceso empírico. Los roles de Dueño de Producto y Scrum Master no cuentan en el cálculo del tamaño del equipo a menos que también estén contribuyendo a trabajar en la Lista de Pendientes de Sprint (Sprint Backlog). (Schwaber & Sutherland, 2013, p.7)

2.2.1.3.3 *El Scrum Master.*

El Scrum Master es el responsable de asegurar que Scrum es entendido y adoptado. Los Scrum Masters hacen esto asegurándose

de que el Equipo Scrum trabaja ajustándose a la teoría, prácticas y reglas de Scrum.

El Scrum Master es un líder que está al servicio del Equipo Scrum. El Scrum Master ayuda a las personas externas al Equipo Scrum a entender qué interacciones con el Equipo Scrum pueden ser de ayuda y cuáles no. El Scrum Master ayuda a todos a modificar estas interacciones para maximizar el valor creado por el Equipo Scrum. (Schwaber & Sutherland, 2013, p.8)

El Servicio del Scrum Master al Dueño de Producto

- Encontrar técnicas para gestionar la Lista de Producto de manera efectiva.
- Ayudar al Equipo Scrum a entender la necesidad de contar con elementos de Lista de Producto claros y concisos.
- Entender la planificación del producto en un entorno empírico.
- Asegurar que el Dueño de Producto conozca cómo ordenar la Lista de Producto para maximizar el valor.
- Entender y practicar la agilidad.
- Facilitar los eventos de Scrum según se requiera o necesite.

El Servicio del Scrum Master al Equipo de Desarrollo

- Guiar al Equipo de Desarrollo en ser auto organizado y multifuncional.
- Ayudar al Equipo de Desarrollo a crear productos de alto valor.

- Eliminar impedimentos para el progreso del Equipo de Desarrollo.
- Facilitar los eventos de Scrum según se requiera o necesite.
- Guiar al Equipo de Desarrollo en el entorno de organizaciones en las que Scrum aún no ha sido adoptado y entendido por completo.

El Servicio del Scrum Master a la Organización

- Liderar y guiar a la organización en la adopción de Scrum.
- Planificar las implementaciones de Scrum en la organización.
- Ayudar a los empleados e interesados a entender y llevar a cabo Scrum y el desarrollo empírico de producto.
- Motivar cambios que incrementen la productividad del Equipo Scrum.
- Trabajar con otros Scrum Masters para incrementar la efectividad de la aplicación de Scrum en la organización.
-

2.2.1.4 Eventos de Scrum.

En Scrum existen eventos predefinidos con el fin de crear regularidad y minimizar la necesidad de reuniones no definidas en Scrum. Todos los eventos son bloques de tiempo (time-boxes), de tal modo que todos tienen una duración máxima. Una vez que comienza un Sprint, su duración es fija y no puede acortarse o alargarse. Los demás eventos pueden terminar siempre que se alcance el objetivo

del evento, asegurando que se emplee una cantidad apropiada de tiempo sin permitir desperdicio en el proceso.

Además del propio Sprint, que es un contenedor del resto de eventos, cada uno de los eventos de Scrum constituye una oportunidad formal para la inspección y adaptación de algún aspecto. Estos eventos están diseñados específicamente para habilitar las vitales transparencia e inspección. La falta de alguno de estos eventos da como resultado una reducción de la transparencia y constituye una oportunidad perdida para inspeccionar y adaptarse. (Schwaber & Sutherland, 2013, p.9)

2.2.1.4.1 El Sprint.

El corazón de Scrum es el Sprint, es un bloque de tiempo (time-box), de un mes o menos durante el cual se crea un incremento de producto “Terminado”, utilizable y potencialmente desplegable. Es más conveniente si la duración de los Sprints es consistente a lo largo del esfuerzo de desarrollo. Cada nuevo Sprint comienza inmediatamente después de la finalización del Sprint previo.

Los Sprints contienen y consisten de la Reunión de Planificación del Sprint (Sprint Planning Meeting), los Scrums Diarios (Daily Scrums), el trabajo de desarrollo, la Revisión del Sprint (Sprint Review), y la Retrospectiva del Sprint (Sprint Retrospective). (Schwaber & Sutherland, 2013, p.9)

Durante el Sprint:

- No se realizan cambios que puedan afectar al Objetivo del Sprint (Sprint Goal).
- Los objetivos de calidad no disminuyen.
- El alcance puede ser clarificado y renegociado entre el Dueño de Producto y el Equipo de Desarrollo a medida que se va aprendiendo más.

Cada Sprint puede considerarse un proyecto con un horizonte no mayor de un mes. Al igual que los proyectos, los Sprints se usan para lograr algo. Cada Sprint tiene una definición de qué se va a construir, un diseño y un plan flexible que guiará la construcción y el trabajo y el producto resultante.

Los Sprints están limitados a un mes calendario. Cuando el horizonte de un Sprint es demasiado grande la definición de lo que se está construyendo podría cambiar, la complejidad podría elevarse y el riesgo podría aumentar. Los Sprints habilitan la predictibilidad al asegurar la inspección y adaptación del progreso al menos en cada mes calendario. Los Sprints también limitan el riesgo al costo de un mes calendario. (Schwaber & Sutherland, 2013, pp.9-10)

Cancelación de un Sprint

Un Sprint puede ser cancelado antes de que el bloque de tiempo llegue a su fin. Solo el Dueño de Producto tiene la autoridad para cancelar el Sprint, aunque puede hacerlo bajo la influencia de los interesados, del Equipo de Desarrollo o del Scrum Master.

Un Sprint se cancelaría si el Objetivo del Sprint llega a quedar obsoleto. Esto podría ocurrir si la compañía cambia la dirección o si las condiciones del mercado o de la tecnología cambian. En general, un Sprint debería cancelarse si no tuviese sentido seguir con él dadas las circunstancias. Pero debido a la corta duración de los Sprints, rara vez la cancelación tiene sentido.

Cuando se cancela un Sprint, se revisan todos los Elementos de la Lista de Producto que se hayan completado y “Terminado”. Si una parte del trabajo es potencialmente entregable, el Dueño de Producto normalmente lo acepta. Todos los Elementos de la Lista de Producto no completados se vuelven a estimar y se vuelven a introducir en la Lista de Producto. El trabajo finalizado en ellos pierde valor con rapidez y frecuentemente debe volverse a estimar.

Las cancelaciones de Sprint consumen recursos, ya que todos deben reagruparse en otra Reunión de Planificación de Sprint para empezar otro Sprint. Las cancelaciones de Sprint son a menudo traumáticas para el Equipo Scrum y son muy poco comunes. (Schwaber & Sutherland, 2013, p.10)

2.2.1.4.2 Reunión de Planificación de Sprint (Sprint Planning Meeting).

El trabajo a realizar durante el Sprint se planifica en la Reunión de Planificación de Sprint. Este plan se crea mediante el trabajo colaborativo del Equipo Scrum completo. La Reunión de Planificación de Sprint tiene un máximo de duración de ocho horas para un Sprint de un mes. Para Sprints más cortos, el evento es

usualmente más corto. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El Scrum Master enseña al Equipo Scrum a mantenerse dentro del bloque de tiempo. (Schwaber & Sutherland, 2013, p.10)

La Reunión de Planificación de Sprint responde a las siguientes preguntas:

- ¿Qué puede entregarse en el Incremento resultante del Sprint que comienza?
- ¿Cómo se conseguirá hacer el trabajo necesario para entregar el Incremento?

Tema Uno: ¿Qué puede ser terminado en este Sprint?

El Equipo de Desarrollo trabaja para proyectar la funcionalidad que se desarrollará durante el Sprint. El Dueño de Producto discute el objetivo que el Sprint debería lograr y los Elementos de la Lista de Producto que, si se completan en el Sprint, lograrían el Objetivo del Sprint. El Equipo Scrum completo colabora en el entendimiento del trabajo del Sprint.

La entrada a esta reunión está constituida por la Lista de Producto, el último Incremento de producto, la capacidad proyectada del Equipo de Desarrollo para el Sprint, y el rendimiento pasado del Equipo de Desarrollo. El número de elementos de la Lista de Producto seleccionados para el Sprint depende únicamente del desarrollo. Solo el Equipo de Desarrollo puede evaluar qué es capaz de lograr durante el Sprint que comienza.

Después de que el Equipo de Desarrollo proyecta qué elementos de la Lista de Producto entregará en el Sprint, el Equipo Scrum elabora un Objetivo del Sprint (Sprint Goal). El Objetivo del Sprint debería lograrse durante el Sprint a través de la implementación de la Lista de Producto, y provee una guía al equipo de desarrollo de porqué se está construyendo el incremento. (Schwaber & Sutherland, 2013, pp.10-11)

Tema Dos: ¿Cómo se conseguirá completar el trabajo seleccionado?

Una vez que se ha establecido el objetivo y seleccionado los elementos de la Lista de Producto para el Sprint, el Equipo de Desarrollo decide cómo construirá esta funcionalidad para formar un Incremento de producto “Terminado”. Los elementos de la Lista de Producto seleccionados para este Sprint, más el plan para terminarlos, recibe el nombre de Lista de Pendientes del Sprint (Sprint Backlog).

El Equipo de Desarrollo por lo general comienza diseñando el sistema y el trabajo necesario para convertir la Lista de Producto en un Incremento de producto funcional. El trabajo podría ser de tamaño o esfuerzo estimado variables. Sin embargo, durante la Reunión de Planificación del Sprint, se planifica suficiente trabajo como para que el Equipo de Desarrollo pueda hacer una proyección de lo que cree que puede completar en el Sprint que comienza. Para el final de esta reunión, el trabajo planificado por el Equipo de Desarrollo para los primeros días del Sprint es descompuesto en unidades de un día o

menos. El Equipo de desarrollo se autoorganiza para asumir el trabajo de la Lista de Pendientes de Sprint, tanto durante la reunión de Planificación de Sprint como a lo largo del Sprint.

El Dueño de Producto puede ayudar a clarificar los elementos de la Lista de Producto seleccionados y hacer concesiones. Si el Equipo de Desarrollo determina que tiene demasiado trabajo o que no tiene suficiente trabajo, podría renegociar los elementos de la Lista de Producto seleccionados con el Dueño de Producto. El Equipo de Desarrollo podría también invitar a otras personas a que asistan con el fin de que proporcionen Asesoría técnica o relacionada con el dominio. Al finalizar la Reunión de Planificación de Sprint, el Equipo de Desarrollo debería ser capaz de explicar al Dueño de Producto y al Scrum Master cómo pretende trabajar como un equipo autoorganizado para lograr el Objetivo del Sprint y crear el Incremento esperado. (Schwaber & Sutherland, 2013, p.11)

Objetivo del Sprint (Sprint Goal)

El Objetivo del Sprint es una meta establecida para el Sprint que puede ser alcanzada mediante la implementación de la Lista de Producto. Proporciona una guía al Equipo de Desarrollo acerca de por qué está construyendo el incremento. Es creado durante la reunión de Planificación del Sprint. El objetivo del Sprint ofrece al equipo de desarrollo cierta flexibilidad con respecto a la funcionalidad implementada en el Sprint. Los elementos de la Lista del Producto seleccionados ofrecen una función coherente, que

puede ser el objetivo del Sprint. El objetivo del Sprint puede representar otro nexo de unión que haga que el Equipo de Desarrollo trabaje en conjunto y no en iniciativas separadas.

A medida que el equipo de desarrollo trabaja, se mantiene el objetivo del Sprint en mente. Con el fin de satisfacer el objetivo del Sprint se implementa la funcionalidad y la tecnología. Si el trabajo resulta ser diferente de lo que el Equipo de Desarrollo espera, ellos colaboran con el Dueño del Producto para negociar el alcance de la Lista de pendientes del Sprint (Sprint Backlog). (Schwaber & Sutherland, 2013, p.12)

2.2.1.4.3 *Scrum Diario (Daily Scrum).*

El Scrum Diario es una reunión con un bloque de tiempo de 15 minutos para que el Equipo de Desarrollo sincronice sus actividades y cree un plan para las siguientes 24 horas. Esto se lleva a cabo inspeccionando el trabajo avanzado desde el último Scrum Diario y haciendo una proyección acerca del trabajo que podría completarse antes del siguiente. El Scrum Diario se realiza a la misma hora y en el mismo lugar todos los días para reducir la complejidad. (Schwaber & Sutherland, 2013, p.12)

Durante la reunión, cada miembro del Equipo de Desarrollo

explica:

- ¿Qué hice ayer que ayudó al Equipo de Desarrollo a lograr el Objetivo del Sprint?

- ¿Qué haré hoy para ayudar al Equipo de Desarrollo a lograr el Objetivo del Sprint?
- ¿Veó algún impedimento que evite que el Equipo de Desarrollo o yo logremos el Objetivo del Sprint?

El Equipo de Desarrollo usa el Scrum Diario para evaluar el progreso hacia el Objetivo del Sprint y para evaluar qué tendencia sigue este progreso hacia la finalización del trabajo contenido en la Lista del Sprint. El Scrum Diario optimiza las posibilidades de que el Equipo de Desarrollo cumpla el Objetivo del Sprint. Cada día, el Equipo de Desarrollo debería entender cómo intenta trabajar en conjunto como un equipo autoorganizado para lograr el Objetivo del Sprint y crear el Incremento esperado hacia el final del Sprint. El Equipo de Desarrollo o los miembros del equipo a menudo se vuelven a reunir inmediatamente después del Scrum Diario, para tener discusiones detalladas, o para adaptar, o replanificar el resto del trabajo del Sprint.

El Scrum Master se asegura de que el Equipo de Desarrollo tenga la reunión, pero el Equipo de Desarrollo es el responsable de dirigir el Scrum Diario. El Scrum Master Enseña al Equipo de Desarrollo para que mantenga el Scrum Diario en los límites del bloque de tiempo de 15 minutos.

El Scrum Master se asegura de que se cumpla la regla de que solo los miembros del Equipo de Desarrollo participan en el Scrum Diario.

Los Scrum Diarios mejoran la comunicación, eliminan la necesidad de mantener otras reuniones, identifican y eliminan impedimentos relativos al desarrollo, resaltan y promueven la toma de decisiones rápida, y mejoran el nivel de conocimiento del Equipo de Desarrollo. El Scrum Diario constituye una reunión clave de inspección y adaptación. (Schwaber & Sutherland, 2013, pp.12-13)

2.2.1.4.4 *Revisión de Sprint (Sprint Review).*

Al final del Sprint se lleva a cabo una Revisión de Sprint para inspeccionar el Incremento y adaptar la Lista de Producto si fuese necesario. Durante la Revisión de Sprint, el Equipo Scrum y los interesados colaboran acerca de lo que se hizo durante el Sprint. Basándose en esto, y en cualquier cambio a la Lista de Producto durante el Sprint, los asistentes colaboran para determinar las siguientes cosas que podrían hacerse para optimizar el valor. Se trata de una reunión informal, no una reunión de seguimiento, y la presentación del Incremento tiene como objetivo facilitar la retroalimentación de información y fomentar la colaboración.

Se trata de una reunión restringida a un bloque de tiempo de cuatro horas para Sprints de un mes. Para Sprints más cortos, se reserva un tiempo proporcionalmente menor. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El Scrum Master enseña a todos a mantener el evento dentro del bloque de tiempo fijado. (Schwaber & Sutherland, 2013, p.13)

La Revisión de Sprint incluye los siguientes elementos:

- Los asistentes son el Equipo Scrum y los interesados clave invitados por el Dueño de Producto
- El Dueño de Producto explica qué elementos de la Lista de Producto se han “Terminado” y cuales no se han “Terminado”
- El Equipo de Desarrollo habla acerca de qué fue bien durante el Sprint, qué problemas aparecieron y cómo fueron resueltos esos problemas
- El Equipo de Desarrollo demuestra el trabajo que ha “Terminado” y responde preguntas acerca del Incremento
- El Dueño de Producto habla acerca de la Lista de Producto en el estado actual. Proyecta fechas de finalización probables en el tiempo basándose en el progreso obtenido hasta la fecha (si es necesario)
- El grupo completo colabora acerca de qué hacer a continuación, de modo que la Revisión del Sprint proporcione información de entrada valiosa para Reuniones de Planificación de Sprints subsiguientes
- Revisión de cómo el mercado o el uso potencial del producto podría haber cambiado lo que es de más valor para hacer a continuación
- Revisión de la línea de tiempo, presupuesto, capacidades potenciales y mercado para la próxima entrega prevista del producto

El resultado de la Revisión de Sprint es una Lista de Producto revisada, que define los elementos de la Lista de Producto posibles para el siguiente Sprint. Es posible además que la Lista de Producto reciba un ajuste general para enfocarse en nuevas oportunidades. (Schwaber & Sutherland, 2013, pp.13-14)

2.2.1.4.5 *Retrospectiva de Sprint (Sprint Retrospective).*

La Retrospectiva de Sprint es una oportunidad para el Equipo Scrum de inspeccionarse a sí mismo y crear un plan de mejoras que sean abordadas durante el siguiente Sprint.

La Retrospectiva de Sprint tiene lugar después de la Revisión de Sprint y antes de la siguiente Reunión de Planificación de Sprint. Se trata de una reunión restringida a un bloque de tiempo de tres horas para Sprints de un mes. Para Sprints más cortos se reserva un tiempo proporcionalmente menor. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El Scrum Master enseña a todos a mantener el evento dentro del bloque de tiempo fijado. El Scrum Master participa en la reunión como un miembro del equipo ya que la responsabilidad del proceso Scrum recae sobre él. (Schwaber & Sutherland, 2013, p.14)

El propósito de la Retrospectiva de Sprint es:

- Inspeccionar cómo fue el último Sprint en cuanto a personas, relaciones, procesos y herramientas

- Identificar y ordenar los elementos más importantes que salieron bien y las posibles mejoras
- Crear un plan para implementar las mejoras a la forma en la que el Equipo Scrum desempeña su trabajo

El Scrum Master alienta al equipo para que mejore, dentro del marco de proceso Scrum, su proceso de desarrollo y sus prácticas para hacerlos más efectivos y amenos para el siguiente Sprint. Durante cada Retrospectiva de Sprint, el Equipo Scrum planifica formas de aumentar la calidad del producto mediante la adaptación de la Definición de “Terminado” (Definition of “Done”) según sea conveniente.

Para el final de la Retrospectiva de Sprint, el Equipo Scrum debería haber identificado mejoras que implementará en el próximo Sprint. El hecho de implementar estas mejoras en el siguiente Sprint, constituye la adaptación subsecuente a la inspección del Equipo de Desarrollo a sí mismo. Aunque las mejoras pueden implementarse en cualquier momento, la Retrospectiva de Sprint ofrece un evento dedicado para este fin, enfocado en la inspección y la adaptación. (Schwaber & Sutherland, 2013, p.14)

2.2.1.5 Artefactos de SCRUM.

Los artefactos de Scrum representan trabajo o valor en diversas formas que son útiles para proporcionar transparencia y oportunidades para la inspección y adaptación. Los artefactos definidos por Scrum están diseñados específicamente para maximizar la transparencia de

la información clave, que es necesaria para asegurar que todos tengan el mismo entendimiento del artefacto. (Schwaber & Sutherland, 2013, p.15)

Historias de Usuario

Mike Cohn (2004), citado por (Menzinsky, López, & Palacio, 2016), sostiene que: Las historias de usuario son utilizadas en los métodos ágiles para la especificación de requisitos, son una descripción breve de una funcionalidad software tal y como la percibe el usuario. Describen lo que el cliente o el usuario quiere que se implemente y se escriben con una o dos frases utilizando el lenguaje común del usuario. Cada historia de usuario debe ser limitada, esta debería poderse memorizar fácilmente y escribir sobre una tarjeta o post-it (card). Las historias de usuario poseen varios campos y entre los que se consideran más necesarios están el ID, como identificador único; el título de la historia de usuario; la descripción, que sintetiza la historia de usuario; la estimación del tiempo de implementación y la prioridad, que permite determinar el orden en que las historias de usuario deben ser implementadas. A continuación, en la tabla 1 se presenta un ejemplo de historia de usuario.

Tabla 1
Ejemplo de Historia de Usuario

HISTORIA DE USUARIO 1	MOCKUPS
<p>Como <i>cliente (rol de usuario)</i></p> <p>Quiero <i>ver un bosquejo de las interfaces con las que el estudiante interactuará al usar el aplicativo (Objetivo)</i></p> <p>Para poder <i>tener una idea de cómo se verá el prototipo (finalidad)</i></p>	
Prioridad: 1	Estimación: 5 h
Criterios de aceptación: Imágenes de cada interfaz a modo de mockups	

Fuente: Elaboración propia

2.2.1.5.1 Lista de Producto (Product Backlog).

La Lista de Producto es una lista ordenada de todo lo que podría ser necesario en el producto, y es la única fuente de requisitos para cualquier cambio a realizarse en el producto. El Dueño de Producto (Product Owner) es el responsable de la Lista de Producto, incluyendo su contenido, disponibilidad y ordenación.

Una Lista de Producto nunca está completa. El desarrollo más temprano de la misma solo refleja los requisitos conocidos y mejor entendidos al principio. La Lista de Producto evoluciona a medida que el producto y el entorno en el que se usará también lo hacen.

La Lista de Producto es dinámica; cambia constantemente para identificar lo que el producto necesita para ser adecuado, competitivo y útil. Mientras el producto exista, su Lista de Producto también existe.

La Lista de Producto enumera todas las características, funcionalidades, requisitos, mejoras y correcciones que constituyen

cambios a ser hechos sobre el producto para entregas futuras. Los elementos de la Lista de Producto tienen como atributos la descripción, la ordenación, la estimación y el valor.

A medida que un producto es utilizado y se incrementa su valor, y el mercado proporciona retroalimentación, la Lista de Producto se convierte en una lista más larga y exhaustiva. Los requisitos nunca dejan de cambiar, así que la Lista de Producto es un artefacto vivo. Los cambios en los requisitos de negocio, las condiciones del mercado o la tecnología podrían causar cambios en la Lista de Producto.

A menudo, varios Equipos Scrum trabajan juntos en el mismo producto. Para describir el trabajo a realizar sobre el producto, se utiliza una única Lista de Producto. En ese caso podría emplearse un atributo de la Lista de Producto para agrupar los elementos.

El refinamiento (refinement) de la Lista de Producto es el acto de añadir detalle, estimaciones y ordena los elementos de la Lista de Producto. Se trata de un proceso continuo, en el cual el Dueño de Producto y el Equipo de Desarrollo colaboran acerca de los detalles de los elementos de la Lista de Producto. Durante el refinamiento de la Lista de Producto, se examinan y revisan sus elementos. El Equipo Scrum decide cómo y cuándo se hace el refinamiento. Este usualmente consume no más del 10% de la capacidad del Equipo de Desarrollo. Sin embargo, los elementos de la Lista de Producto pueden actualizarse en cualquier momento por el Dueño de Producto o a criterio suyo.

Los elementos de la Lista de Producto de orden más alto son generalmente más claros y detallados que los de menor orden. Se realizan estimaciones más precisas basándose en la mayor claridad y detalle; cuanto más bajo es el orden, menor es el detalle. Los elementos de la Lista de Producto de los que se ocupará el Equipo de Desarrollo en el siguiente Sprint tienen una granularidad mayor, habiendo sido descompuestos de forma que cualquier elemento puede ser “Terminado” dentro de los límites del bloque de tiempo del Sprint. Los elementos de la Lista de Producto que pueden ser “Terminados” por el Equipo de Desarrollo en un Sprint son considerados “preparados” o “accionables” para ser seleccionados en una reunión de Planificación de Sprint. Los elementos de la Lista de Producto normalmente adquieren este grado de transparencia mediante las actividades de refinamiento descritas anteriormente.

El Equipo de Desarrollo es el responsable de proporcionar todas las estimaciones. El Dueño de Producto podría influenciar al Equipo ayudándoles a entender y seleccionar soluciones de compromiso, pero las personas que harán el trabajo son las que hacen la estimación final. (Schwaber & Sutherland, 2013, pp.15-16)

Formato de la lista del producto

Scrum prefiere la comunicación verbal o de visualización directa, a la escrita. La pila del producto no es un documento de requisitos, sino una herramienta de información para el equipo. (Menzinsky et al., 2016, p.23)

Si se emplea formato de lista, la información mínima que se suele incluir para cada historia de usuario es:

- Descripción de la funcionalidad/requisito, denominado “historia de usuario”
- Prioridad
- Preestimación del esfuerzo necesario

Y a veces también un código o identificador único de la historia.

Por las características del proyecto o del equipo, se pueden incluir en la pila del producto información adicional como:

- Observaciones.
- Criterio de validación.
- Persona asignada.
- N.º de Sprint en el que se realiza.
- Módulo del sistema al que pertenece.

A continuación, en la tabla 2 se presenta un ejemplo del Product Backlog.

Tabla 2
Ejemplo de Product Backlog

Código	Requerimiento	Estimación (horas)	Prioridad
RQ-01	Desarrollo de mockups	5	1
RQ-02	Modelado de la base de datos	5	1
RQ-03	Creación de la base de datos	5	1

Fuente: Elaboración propia

2.2.1.5.2 *Lista de Pendientes del Sprint (Sprint Backlog).*

La Lista de Pendientes del Sprint es el conjunto de elementos de la Lista de Producto seleccionados para el Sprint, más un plan para entregar el Incremento de producto y conseguir el Objetivo del Sprint. La Lista de Pendientes del Sprint es una predicción hecha por el Equipo de Desarrollo acerca de qué funcionalidad formará parte del próximo Incremento y del trabajo necesario para entregar esa funcionalidad en un Incremento “Terminado”. La Lista de Pendientes del Sprint hace visible todo el trabajo que el Equipo de Desarrollo identifica como necesario para alcanzar el Objetivo del Sprint.

La Lista de Pendientes del Sprint es un plan con un nivel de detalle suficiente como para que los cambios en el progreso se puedan entender en el Scrum Diario. El Equipo de Desarrollo modifica la Lista de Pendientes del Sprint durante el Sprint y esta Lista de Pendientes del Sprint emerge a lo largo del Sprint. Esto ocurre a medida que el Equipo de Desarrollo trabaja sobre el plan y aprende más acerca del trabajo necesario para conseguir el Objetivo del Sprint.

Según se requiere nuevo trabajo, el Equipo de Desarrollo lo añade a la Lista de Pendientes del Sprint. A medida que el trabajo se ejecuta o se completa, se va actualizando la estimación de trabajo restante. Cuando algún elemento del plan pasa a ser considerado innecesario, es eliminado. Solo el Equipo de Desarrollo puede cambiar su Lista de Pendientes del Sprint durante un Sprint. La Lista de Pendientes del Sprint es una imagen visible en tiempo real del

trabajo que el Equipo de Desarrollo planea llevar a cabo durante el Sprint, y pertenece únicamente al Equipo de Desarrollo. (Schwaber & Sutherland, 2013, pp.16-17)

La lista de pendientes del sprint descompone las historias de usuario en unidades de tamaño adecuado para monitorizar el avance a diario, e identificar riesgos y problemas sin necesidad de procesos de gestión complejos. Es también una herramienta para la comunicación visual directa del equipo. (Menzinsky et al., 2016, p.24)

Condiciones

- Realizada de forma conjunta por todos los miembros del equipo.
- Cubre todas las tareas identificadas por el equipo para conseguir el objetivo del sprint.
- Sólo el equipo la puede modificar durante el sprint.
- Las tareas demasiado grandes deben descomponerse en otras más pequeñas. En ningún caso una tarea puede tener un tamaño tal que necesite más de un día de trabajo.
- Es visible para todo el equipo. Idealmente en un tablero o pared en el mismo espacio físico donde trabaja el equipo.

Formato de Lista de Pendientes del Sprint:

Lo más adecuado a las características del proyecto, oficina y equipo, lo apropiado es diseñar el formato más cómodo para todos, teniendo en cuenta los siguientes criterios:

- Incluir la siguiente información: Lista de Pendientes del sprint, persona responsable de cada tarea, estado en el que se encuentra y tiempo de trabajo que queda para completarla.
- Incluir sólo la información estrictamente necesaria.
- Debe servir de medio para registrar en cada reunión diaria del sprint, el tiempo que le queda a cada tarea.
- Facilitar la consulta y la comunicación diaria y directa del equipo.

Durante el sprint, el equipo actualiza a diario en ella los tiempos pendientes de cada tarea. (Menzinsky et al., 2016, p.25)

A continuación, en la tabla 3 se presenta un ejemplo del Sprint Backlog.

Tabla 3
Ejemplo de Sprint Backlog

Iteración	Código	Requerimiento
Sprint 1	RQ-01	Desarrollo de mockups
	RQ-02	Modelado de la base de datos
	RQ-03	Creación de la base de datos
Sprint 2	RQ-04	Maquetado del Home
	RQ-05	Maquetado de la sección mi cuenta

Fuente: Elaboración propia

Seguimiento del Progreso del Sprint

En cualquier momento durante un Sprint, es posible sumar el trabajo restante total en los elementos de la Lista de Pendientes del Sprint. El Equipo de Desarrollo hace seguimiento de este trabajo restante total al menos en cada Scrum Diario para proyectar la posibilidad de conseguir el Objetivo del Sprint. Haciendo seguimiento del trabajo restante a lo largo del Sprint, el Equipo de Desarrollo puede gestionar su progreso. (Schwaber & Sutherland, 2013, p.17)

Transparencia de los Artefactos

Scrum se basa en la transparencia. Las decisiones para optimizar el valor y controlar el riesgo se toman basadas en el estado percibido de los artefactos. En la medida en que la transparencia sea completa, estas decisiones tienen unas bases sólidas. En la medida en que los artefactos no son completamente transparentes, estas decisiones pueden ser erróneas, el valor puede disminuir y el riesgo puede aumentar.

El Scrum Master debe trabajar con el Dueño de Producto, el Equipo de Desarrollo y otras partes involucradas para entender si los artefactos son completamente transparentes. Hay prácticas para hacer frente a la falta de transparencia; el Scrum Master debe ayudar a todos a aplicar las prácticas más apropiadas si no hay una transparencia completa. Un Scrum Master puede detectar la falta de transparencia inspeccionando artefactos, reconociendo patrones,

escuchando atentamente lo que se dice y detectando diferencias entre los resultados esperados y los reales. La labor del Scrum Master es trabajar con el Equipo Scrum y la organización para mejorar la transparencia de los artefactos. Este trabajo usualmente incluye aprendizaje, convicción y cambio. La transparencia no ocurre de la noche a la mañana, sino que es un camino. (Schwaber & Sutherland, 2013, pp.17-18)

Incremento

El Incremento es la suma de todos los elementos de la Lista de Producto completados durante un Sprint y el valor de los incrementos de todos los Sprints anteriores. Al final de un Sprint, el nuevo Incremento debe estar “Terminado”, lo cual significa que está en condiciones de ser utilizado y que cumple la Definición de “Terminado” del Equipo Scrum. El incremento debe estar en condiciones de utilizarse sin importar si el Dueño de Producto decide liberarlo o no. (Schwaber & Sutherland, 2013, p.17)

Definición de “Terminado” (Definition of “Done”)

Cuando un elemento de la Lista de Producto o un Incremento se describe como “Terminado”, todo el mundo debe entender lo que significa “Terminado”. Aunque esto varía significativamente para cada Equipo Scrum, los miembros del Equipo deben tener un entendimiento compartido de lo que significa que el trabajo esté completado, para asegurar la transparencia. Esta es la definición de

“Terminado” para el Equipo Scrum y se utiliza para evaluar cuándo se ha completado el trabajo sobre el Incremento de producto.

Esta misma definición guía al Equipo de Desarrollo en saber cuántos elementos de la Lista de Producto puede seleccionar durante una reunión de Planificación de Sprint. El propósito de cada Sprint es entregar Incrementos de funcionalidad que potencialmente se puedan poner en producción, y que se ajustan a la Definición de “Terminado” actual del Equipo Scrum.

Los Equipos de Desarrollo entregan un Incremento de funcionalidad de producto en cada Sprint. Este Incremento es utilizable, de modo que el Dueño de Producto podría elegir liberarlo inmediatamente. Si la definición de “Terminado” para un incremento es parte de las convenciones, estándares o guías de la organización de desarrollo, al menos todos los Equipos Scrum deben seguirla. Si “Terminado” para un incremento no es una convención de la organización de desarrollo, el Equipo de Desarrollo del Equipo Scrum debe definir una definición de “Terminado” apropiada para el producto.

Si hay múltiples Equipos Scrum trabajando en la entrega del sistema o producto, los equipos de desarrolladores en todos los Equipos Scrum deben definir en conjunto la definición de “Terminado”.

Cada Incremento se integra con todos los Incrementos anteriores y es probado exhaustivamente, asegurando que todos los Incrementos funcionan en conjunto.

A medida que los Equipos Scrum maduran, se espera que su definición de “Terminado” se amplíe para incluir criterios más rigurosos para una mayor calidad. Cualquier producto o sistema debería tener una definición de “Terminado” que es un estándar para cualquier trabajo realizado sobre él. (Schwaber & Sutherland, 2013, p.18)

En la figura 1 se presenta el proceso de Scrum y las relaciones entre sus componentes.

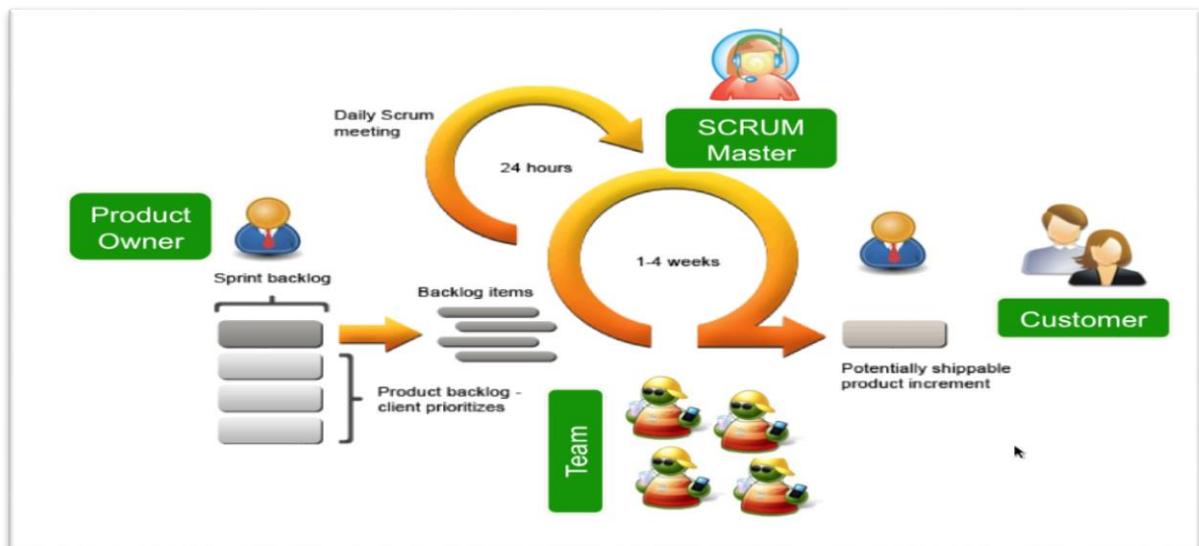


Figura 1. Proceso SCRUM

Fuente: (Caudevilla, Metodología Ágil: SCRUM, 2010)

2.2.2 Tipos de aplicaciones móviles según su desarrollo.

El principal reto para los proveedores de aplicaciones es proporcionar soluciones para todas las plataformas, pero tiene un alto costo.

La solución ideal a este problema es crear y mantener una única aplicación para todas las plataformas. El desarrollo multiplataforma tiene como objetivo mantener la misma base de código para diversas

plataformas. De esta forma el esfuerzo y costo de desarrollo se reduce notablemente. (Delía, Galdamez, Thomas, & Pesado, 2013, p.767)

A continuación, se presentan tres enfoques para desarrollo de aplicaciones para dispositivos móviles: un enfoque nativo y dos enfoques multiplataforma (web e híbrido).

2.2.2.1 Las Apps Nativas.

Las aplicaciones nativas son aquellas que se conciben para ejecutarse en una plataforma específica, es decir, se debe considerar el tipo de dispositivo, el sistema operativo a utilizar y su versión.

El código fuente se compila para obtener código ejecutable, proceso similar que el utilizado para las tradicionales aplicaciones de escritorio.

Cuando la aplicación está lista para ser distribuida debe ser transferida a las App Stores (tiendas de aplicaciones) específicas de cada sistema operativo. Estas tienen un proceso de auditoría para evaluar si la aplicación se adecúa a los requerimientos de la plataforma a operar. Cumplido este paso, la aplicación se pone a disposición de los usuarios.

La principal ventaja de este tipo de aplicaciones es la posibilidad de interactuar con todas las capacidades del dispositivo (cámara, GPS, acelerómetro, agenda, entre otras). Además, no es estrictamente necesario poseer acceso a internet. Su ejecución es rápida, puede ejecutarse en modo background y notificar al usuario cuando ocurra un evento que necesite su atención.

Claramente estas ventajas se pagan con un mayor costo de desarrollo, pues se debe utilizar un lenguaje de programación diferente según la plataforma. Por ende, si se desea cubrir varias plataformas, se deberá generar una aplicación para cada una de ellas. Esto conlleva a mayores costos de actualización y distribución de nuevas versiones. (Delía, Galdamez, Thomas, & Pesado, 2013, p.768)

2.2.2.2 Las Web Apps.

Las aplicaciones web para móviles son diseñadas para ser ejecutadas en el navegador del dispositivo móvil. Estas aplicaciones son desarrolladas utilizando HTML, CSS y JavaScript, es decir, la misma tecnología que la utilizada para crear sitios web.

Una de las ventajas de este enfoque es que los dispositivos no necesitan la instalación de ningún componente en particular, ni la aprobación de algún fabricante para que las aplicaciones sean publicadas y utilizadas. Solo se requiere acceso a internet. Además, las actualizaciones de la aplicación son visualizadas directamente en el dispositivo, ya que los cambios son aplicados sobre el servidor y están disponibles de inmediato. En resumen, es rápido y fácil de poner en marcha.

La principal ventaja de este tipo de aplicación es su independencia de la plataforma. No necesita adecuarse a ningún entorno operativo. Solo es necesario un navegador.

Por contrapartida, esto disminuye la velocidad de ejecución y podrían llegar a ser menos atractivas que las aplicaciones nativas.

Además, podrían tener baja performance por problemas de conectividad. Finalmente, este tipo de aplicaciones no pueden utilizar todos los elementos de hardware del dispositivo, como, por ejemplo, cámara, GPS, entre otros. (Delía, Galdamez, Thomas, & Pesado, 2013, pp.767-768)

2.2.2.3 Las Apps Híbridas

Las aplicaciones híbridas combinan lo mejor de los dos tipos de aplicaciones anteriores. Se utilizan tecnologías multiplataforma como HTML, JavaScript y CSS, pero se puede acceder a buena parte de las capacidades específicas de los dispositivos.

En resumen, son desarrolladas utilizando tecnología web y son ejecutadas dentro de un contenedor web sobre el dispositivo móvil.

Entre las principales ventajas de esta metodología se pueden mencionar la posibilidad de distribución de la aplicación a través de las tiendas de aplicaciones, la reutilización de código para múltiples plataformas y la posibilidad de utilizar las características de hardware del dispositivo.

Una de las desventajas es que, al utilizar la misma interfaz para todas las plataformas, la apariencia de la aplicación no será como la de una aplicación nativa. Finalmente, la ejecución será más lenta que la ejecución en una aplicación nativa. (Delía, Galdamez, Thomas, & Pesado, 2013, pp.768-769)

2.2.3 Aplicaciones web progresivas

Las Progressive Web Apps son experiencias que combinan lo mejor de la Web y lo mejor de las apps. Están disponibles para los usuarios a partir de la primera visita en una pestaña del navegador y no requieren instalación. A medida que el usuario compila progresivamente una relación con la app con el paso del tiempo, se hace más y más poderosa. Se carga rápidamente, incluso con redes débiles, envía notificaciones push relevantes, tiene un ícono en la pantalla principal y se carga como experiencia de pantalla completa y de primer nivel. (LePage, 2018)

Una Progressive Web App es:

- **Progresiva:** Funciona para todos los usuarios, sin importar la elección de navegador, porque está construida con mejora progresiva como principio central.
- **Adaptable:** Se adapta a cualquier factor de formulario, sea escritorio, móvil, tablet o lo que venga en el futuro.
- **Independiente de la conectividad:** Mejorada con service workers para trabajar sin conexión o con redes de mala calidad.
- **Estilo app:** Al usuario le parece una app con interacciones y navegación estilo app, porque está construida con modelo de shell de app.
- **Fresca:** Siempre actualizada gracias al proceso de actualización de service worker.

- **Segura:** Emitida vía HTTPS para evitar intromisiones y para garantizar que el contenido no se haya manipulado.
- **Descubrible:** Se puede identificar como "app" gracias al manifiesto W3C y al alcance de registro de service worker, lo que permite que los motores de búsqueda la encuentren.
- **Posibilidad de volver a interactuar:** Facilita la posibilidad de volver a interactuar a través de funciones como notificaciones push.
- **Instalable:** les permite a los usuarios "conservar" las apps que les resultan más útiles en su pantalla principal sin la molestia de una tienda de app.
- **Vinculable:** se puede compartir fácilmente vía URL, no requiere instalación compleja.

2.2.4 Manejo de control de versiones – GIT.

Es un sistema de control de versiones distribuido diseñado para trabajar con proyectos de diferentes tamaños, haciendo foco en la eficiencia (velocidad de trabajo). Cada directorio de trabajo de Git es un repositorio en sí mismo con un historial completo de las modificaciones realizadas y las capacidades de tracking completas que no depende del acceso a una red o servidor central. (Salamon, A., Maller, P. A., Boggio, A., Mira, N., Perez, S., & Coenda, F., 2014, p.5)

2.2.5 Ionic.

Ionic es una herramienta, gratuita y open source, para el desarrollo de aplicaciones híbridas basadas en HTML5, CSS y JS. Está construido con Sass y optimizado con AngularJS. (Perez Rivas, 2015)

Principales características:

- Alto rendimiento, Ionic está construido para ser rápido gracias a la mínima manipulación del DOM, con cero jQuery y con aceleraciones de transiciones por hardware.
- AngularJS, Ionic utiliza AngularJS con el fin de crear un marco más adecuado para desarrollar aplicaciones ricas y robustas.
- Centro nativo, Ionic se inspira en las SDK de desarrollo móviles nativos más populares
- Bonito diseño, Limpio, sencillo y funcional. Ionic ha sido diseñado para poder trabajar con todos los dispositivos móviles actuales. Con muchos componentes usados en móviles, tipografía, elementos interactivos, etc.
- Un potente CLI, Con un sólo comando podrás crear, construir, probar y compilar tus aplicaciones en cualquier plataforma

2.3 Definición de Términos Básicos

2.3.1 Aplicación web.

Son aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una intranet mediante un navegador (López, 2015).

2.3.2 Arquitectura de una aplicación web.

La arquitectura de una aplicación web requiere una interfaz de usuario la cual se instala y corre en una computadora o estación de trabajo y envía solicitudes a un servidor para ejecutar operaciones complejas (Arquitectura de las aplicaciones Web, 2015).

2.3.3 Base de Datos.

Una base de datos es una colección de datos relacionados. Con la palabra datos nos referimos a los hechos (datos) conocidos que se pueden grabar y que tienen un significado implícito (Elmasri y shamkant, 2007, citado por Pérez, 2012).

2.3.4 Bolsa de trabajo.

Es un lugar físico o virtual donde se ordenan ofertas para que tanto ofertantes como demandantes de empleo puedan encontrar de manera rápida y organizada la información que requieren (Herramientas de búsqueda de empleo, 2013).

2.3.5 Cliente / Servidor web.

El cliente es un programa que requiere o solicita servicios, mientras que un servidor es un programa que brinda servicios (base de datos, imágenes, archivos de texto, correo, etc.). Desde un punto de vista lógico, el cliente y servidor se encuentran separados y la comunicación entre ellos se realiza a través de una infraestructura de red. En entornos web, el cliente realiza las peticiones de un servicio

hacia el servidor el cual responde transfiriendo documentos hipertexto, siendo necesario el uso del protocolo HTTP (Hypertext Transfer Protocol) para que la comunicación se realice (Departamento de Informática de la Universidad de Valladolid, 2006).

2.3.6 CSS3.

CSS es el acrónimo de Cascading Style Sheet; este también un lenguaje de marcado que nos permite aplicar estilos a nuestros elementos HTML. Una novedad de CSS3 frente a las versiones anteriores es que se permite el uso de fuentes personalizadas. El repositorio de fuentes más popular es Google Fonts que ofrece un amplio número de ellas de uso libre (Jiménez Ortega, 2017).

2.3.7 Diseño web adaptable.

El Diseño web adaptable o Responsive Web Design (RWD) comprende una serie de técnicas y pautas de diseño que permiten adaptar sitios web al entorno de navegación del usuario, entendiendo como entorno de navegación la multiplicidad de dispositivos, móviles o no, por medio de los cuales los usuarios pueden acceder y navegar en internet Este concepto surge en el año 2010 de la mano del diseñador web norteamericano Ethan Marcotte (Marcotte, 2010, 2011), sin embargo los orígenes se remontan al surgimiento y atomización de los dispositivos móviles. En el año 2008 el consorcio internacional W3C, que da difusión a pautas, recomendaciones y buenas prácticas para el diseño y mantenimiento de páginas web,

planteaba una serie de recomendaciones específicas para la web en móvil, utilizando el nombre de "One Web", definiendo el camino al que debería apuntar el diseño web para múltiples dispositivos. "One web" se refería a la creación de una web para todos englobando contenidos, experiencia de usuario y accesibilidad a todos los dispositivos disponibles en el mercado (desktop, tabletas, smartphones, consolas...) (González & Marcos Mora, 2013, p.2).

2.3.8 Empresa.

Una empresa es una organización, institución o industria, dedicada a actividades con fines económicos o comerciales, para satisfacer las necesidades de bienes y/o servicios de los demandantes (Empresa, 2017).

2.3.9 HTML5.

HTML (Hypertext Markup Language) es un lenguaje de marcado que sirve para definir la estructura y la semántica de una página web. HTML fue creado y es mantenido por una organización sin ánimo de lucro llamada W3C. El W3C es un consorcio formado por más de 400 empresas (entre ellas las que desarrollan los principales navegadores como Google, Microsoft, Mozilla, Apple...), etc.

Desde el consorcio trabajan continuamente en definir cómo debe evolucionar este lenguaje y otros estándares que conforman la web. Posteriormente los fabricantes de navegadores preparan los mismos intentando conseguir que un código funcione igual en todos

los navegadores. Aunque desafortunadamente no siempre es así, cada vez es una realidad más cercana. Por tanto, a lo largo de los años las versiones de HTML han evolucionado: HTML 2.0 (1995), HTML 4.0 (1997), XHTML (2000), HTML5 (2014), etc. con el objetivo de adaptarse a los nuevos tiempos y así dar soporte a nuevas necesidades (estandarización de los sistemas de audio, vídeo, etc.) (Jiménez Ortega, 2017).

2.3.10 Información.

Es un conjunto organizado de datos procesados, que constituyen un mensaje que cambia el estado de conocimiento del sujeto o sistema que recibe dicho mensaje (Stair R., Reynolds G., 2016).

2.3.11 JavaScript.

JavaScript (JS) es un lenguaje de programación, un lenguaje con su propio vocabulario, sintaxis, semántica, expresiones, errores, etc. JavaScript nos permite darle vida a la web, hacerla más dinámica e interactiva, y por tanto mostrar algo más que información de manera estática (Jiménez Ortega, 2017).

- ¿Qué se puede hacer con JavaScript?
- Operaciones matemáticas, lógicas, etc.
- Controlar el flujo del programa.
- Validar formularios.
- Cargar contenidos mediante peticiones HTTP.

- Acceder a información como la versión del navegador, tamaño de la ventana, sistema operativo, localización, etc.

2.3.12 MySQL.

Es un SGBD (Sistema Gestor de Bases de Datos) de código abierto, multiplataforma, fácil de usar, de buena performance y confiabilidad, utilizado por gran cantidad de sistemas cliente-servidor (aplicaciones web mayoritariamente) (MySQL, 2015). Esta desarrollado en C/C++ y entre sus características resaltantes se encuentran la posibilidad de crear y configurar usuarios, permitiendo asignarles distintos permisos o privilegios; facilidad para la importación y exportación de datos, contar con un conjunto de datos diverso, una API disponible para otros lenguajes de programación (Java, PHP, entre otros), etc. (Casillas Santillán, Gibert Ginesta, & Pérez Mora, 2008).

2.3.13 Navegador web.

Programa que se utiliza para acceder a la web, interpreta los lenguajes como HTML, CSS y JavaScript (Pressman, 2018).

2.3.14 Prototipo.

Desde el punto de vista de la informática, un prototipo de un sistema informático, es una muestra más simplificada de un sistema. Un prototipo provee resultados rápidos y nos da un vistazo a grandes rasgos de cómo será el sistema; este puede aceptarse y convertirse

en un sistema completo, o descartarse al no cumplirse las expectativas previstas. Los prototipos suelen ser presentados a los clientes quienes se encargan de validar los resultados (Alegsa, 2010).

2.3.15 Smartphone.

Un Smartphone es un teléfono móvil que incorpora características de una computadora personal. Los smartphones suelen permitir al usuario instalar nuevas aplicaciones, aumentando así sus funcionalidades. Esto es posible porque ejecutan un sistema operativo potente de fondo. Pueden incluir un mini teclado, una pantalla táctil, un lápiz óptico, etc. Los smartphones tienden a incluir acceso a internet, servicios de e-mail, cámara integrada, navegador web, procesador de textos, etc. (Alegsa, 2015).

CAPÍTULO III: DESARROLLO DEL TRABAJO DE SUFICIENCIA PROFESIONAL

3.1 Bolsa de trabajo.

Una bolsa de trabajo se encarga de buscar y dotar de personal a las empresas que las requieran, dicho personal debe de cumplir con los requerimientos de la empresa solicitante, según las especificaciones del puesto de trabajo. Hoy en día las bolsas de trabajo no solo cuentan con una oficina, también pueden brindar servicio por medio de internet, potenciadas por las tecnologías web y móviles. En el caso de las bolsas de trabajo universitarias, su objetivo es buscar empleo para sus estudiantes y egresados, contactándolos con las empresas que buscan practicantes o egresados.

3.1.1 Objetivos.

Contribuir en la formación del estudiante y del egresado de la Universidad Nacional Tecnológica de Lima Sur a través de establecimiento de la relación universidad – alumno – empresa y del desarrollo integral de sus competencias.

Brindar facilidades a los estudiantes para la auto generación de recursos

económicos para solventar sus estudios universitarios, reducir la morosidad y deserción.

3.1.2 Alcance.

Las empresas interesadas en contactarse con alumnos y egresados de la Universidad Nacional Tecnológica de Lima Sur.

3.1.3 Normas y referencias.

- Ley Universitaria 30220.
- Reglamento Académico General.
- Estatuto de la UNTELS

3.1.4 Responsables

Dueño del producto: jefe de la Oficina General de Tecnologías de la Información y Comunicación.

Equipo de desarrollo: Equipo Scrum.

3.1.5 Líneas guía.

Prototipo de bolsa de trabajo:

Empresa:

- La empresa envía la convocatoria laboral a la Oficina General de Servicios Universitarios.
- La Oficina General de Servicios Universitarios envía la convocatoria laboral a la Oficina General de Tecnologías de la Información y Comunicación.

- La Oficina General de Tecnologías de la Información y Comunicación, publica la convocatoria laboral en el prototipo de bolsa de trabajo.

Estudiante:

- El estudiante inicia sesión en el aplicativo con el mismo usuario y contraseña empleado en el Sistema Integrado de Gestión Universitaria.
- El estudiante creará y subirá su curriculum vitae al aplicativo.
- El estudiante visualizará las ofertas laborales de interés y postulará a la vacante.

Siguiendo los lineamientos de la metodología Scrum se procede a desarrollar sus fases y procesos.

3.2 Fase de Inicio.

3.2.1 Creación de la visión del proyecto.

3.2.1.1 Entrada.

3.2.1.1.1 Caso de negocio del proyecto.

El proyecto se inicia con la presentación del caso de negocio del proyecto a los socios y patrocinadores, a fin de que comprendan los beneficios que se obtendrán y que los patrocinadores confirmen que van a proporcionar los recursos financieros para el proyecto.

Objetivo del negocio.

Desarrollar un prototipo de bolsa de trabajo basado en aplicaciones web progresivas para automatizar los procesos de prácticas pre profesionales y seguimiento al egresado en la Universidad Nacional Tecnológica de Lima Sur.

Justificación del proyecto.

Con el desarrollo del proyecto se contará con un prototipo que permita acceder a información de la bolsa de trabajo desde dispositivos móviles o computadoras de forma fácil y rápida.

3.2.1.1.2 Aseguramiento de la información.

La información es un activo importante del negocio, tiene valor para la organización y por tanto requiere de una protección adecuada. La seguridad de la información protege del impacto de las amenazas y vulnerabilidades, para asegurar la continuidad del negocio, minimizar los daños a la organización, maximizar el retorno de la inversión y las oportunidades de negocio.

a. Evaluación de los riesgos.

En la evaluación de los riesgos se identificaron, cuantificaron y priorizaron los riesgos que pueden afectar al desarrollo del proyecto. A continuación, se muestra los riesgos identificados.

Tabla 4
Riesgos identificados

Código	Descripción del riesgo	Estimación de probabilidad de ocurrencia	Prioridad
R01	Modificación del cronograma de proyecto	3	Alta
R02	Requerimientos no identificados	3	Alta
R03	Cambios en el alcance del proyecto	2	Media
R04	Error en la estimación de costos	2	Media
R05	Incumplimiento con las fechas de entrega	2	Media
R06	Fallas técnicas constantes durante el desarrollo	1	Baja
R07	Deserción de integrantes del equipo de desarrollo	1	Baja

Fuente: Elaboración propia

b. Tratamiento de los riesgos.

Para cada uno de los riesgos identificados, siguiendo la evaluación del riesgo, se necesita elegir el tratamiento del riesgo. Aplicar controles apropiados para reducir riesgos aun nivel aceptable. Selección de controles.

La protección de los datos de carácter personal.

La protección de datos y la privacidad debe ser asegurada como se requiere en la legislación y las regulaciones. Se implemento una política organizacional de privacidad y de protección de datos, esta política fue comunicada a todo el personal implicado en el proyecto.

La salvaguarda de los activos del proyecto.

Se implemento un control de versiones mediante la creación de un repositorio en el servicio de alojamiento Bitbucket en el cual se almacenó el proyecto. Se debe tener cuidado al distribuir la política de seguridad fuera de la organización con el fin de no compartir información confidencial.

La documentación de la política de seguridad de la información.

El documento de política de seguridad de la información fue aprobada y publicada por el propietario del producto, quien lo dio a conocer a todos los involucrados en el proyecto. El documento contiene la siguiente información:

- Definición de seguridad de la información sus objetivos globales, el alcance de la seguridad y su importancia.
- Una breve explicación de las políticas, principios, normas y requisitos de conformidad más importantes para la organización.
- Una definición de las responsabilidades generales y específicas en materia de gestión de la seguridad de la información.

Se debe proteger de los activos importantes del proyecto frente a su pérdida, destrucción o falsificación.

La asignación de responsabilidades de seguridad.

Se definieron claramente las responsabilidades sobre seguridad de la información.

Roles y responsabilidades

Socios:

- Interactúa con el equipo principal de Scrum para brindarle información sobre la gestión de riesgos que afectan el logro de los resultados esperados, así como los beneficios del proyecto.

Propietario del producto:

- Captura y evalúa riesgos del proyecto.
- Prioriza y comunica los riesgos a los socios relevantes y al equipo Scrum.
- Se asegura de que los niveles de riesgo del proyecto estén dentro de los límites aceptables.

Scrum Master:

- Le facilita al equipo Scrum la identificación y escalamiento de riesgos.

Equipo Scrum:

- Identifica los riesgos durante el desarrollo del producto durante el proceso de creación de entregables.

- Implementa actividades de gestión de riesgo tal como lo recomienda el propietario del producto.

Control de accesos.

Se busca controlar los accesos a la información, los procesos del negocio y al aplicativo. Se deberían tener en cuenta para ello las políticas de distribución de la información y de autorizaciones. El control de accesos contiene los siguientes elementos.

- Requisitos de negocio para el control de accesos.
- Gestión de acceso de usuarios.
- Responsabilidades de los usuarios.
- Control de acceso a la aplicación.

Seguridad de la aplicación.

Tiene por objetivo evitar pérdidas, modificaciones o mal uso de los datos de usuario en la aplicación.

Validación de los datos de entrada: Se deben validar los datos de entrada a la aplicación para garantizar que son correctas y apropiadas.

Integridad de mensajes: Se debe asegurar la autenticación y protección de la transmisión de data en la aplicación, mediante encriptado y uso de https.

Validación de los datos de salida: Se deben validar los datos de salida de la aplicación para garantizar que ha sido correcto y apropiado a las circunstancias.

La gestión de la vulnerabilidad técnica.

Se busca reducir los riesgos resultantes de la explotación de vulnerabilidades técnicas.

Control de las vulnerabilidades técnicas: Se debe obtener a tiempo la información sobre las vulnerabilidades técnicas del aplicativo. Igualmente se debe evaluar la exposición de la organización a tales vulnerabilidades y las medidas apropiadas para tratar a los riesgos asociados.

La gestión de las mejoras e incidencias en la seguridad de información.

Las responsabilidades y procedimientos deben establecerse para maniobrar los eventos y debilidades en la seguridad de información de una manera efectiva una vez que hayan sido reportados. Un proceso de mejora continua debe ser aplicado en respuesta al monitoreo, evaluación y gestión general de los incidentes en la seguridad de información.

Responsabilidades y procedimientos: Las responsabilidades y procedimientos de la gerencia deben ser

establecidas para asegurar una rápida, efectiva y ordenada respuesta a los incidentes en la seguridad de información.

Aprendiendo de los incidentes en la seguridad de información: La información ganada de la evaluación de los incidentes en la seguridad de información deben ser utilizados para identificar incidentes que se repiten o de gran impacto.

Recolección de evidencia: Después de un incidente en la seguridad de información, la evidencia debe ser recolectada, retenida y presentada para estar conforme con las reglas para las acciones correctivas, procesos disciplinarios, etc.

3.2.1.2 Herramientas.

3.2.1.2.1 Reuniones de la visión del proyecto.

Se realizó una reunión con el propietario del producto del programa y los socios. En esta se identificaron los requerimientos de negocio y las expectativas de los socios a fin de desarrollar una declaración de la visión del proyecto eficaz.

3.2.1.2.2 Análisis por medio de encuesta.

Para recabar la opinión de los estudiantes respecto a la bolsa de trabajo UNTELS y determinar su interés sobre una aplicación que desempeñe una función similar, pero que aproveche las bondades de las aplicaciones web progresivas, se desarrolló una encuesta online (accesible a partir del siguiente enlace:

<https://goo.gl/forms/sz8yrnlsXutXSzli2>), esta encuesta se encuentra en el anexo 1.

3.2.1.3 Salidas.

3.2.1.3.1 *Propietario del producto identificado.*

Propietario del producto: Denisse Marie Barrientos Pichingue.

3.2.1.3.2 *Declaración de la visión del proyecto.*

Visión del proyecto: Desarrollar un prototipo de bolsa de trabajo con aplicaciones web progresivas que permita mejorar el proceso de prácticas pre profesionales y seguimiento al egresado en la UNTELS.

3.2.1.3.3 *Presupuesto del proyecto.*

Tabla 5
Coste de recursos humanos

N°	Descripción	Recursos	Costo
1	Diseñador	1	S/ 2,000.00
2	Desarrollador Front-End	1	S/ 2,500.00
3	Desarrollador Back-End	1	S/ 3,000.00
4	Scrum master	1	S/ 4,000.00
Total		4	S/ 11,500.00

Fuente: Elaboración propia

Tabla 6
Coste de recursos hardware y software

N°	Descripción	Recursos	Costo anual
1	Servidor web	1	S/ 0.00 *
2	Conexión a internet 20Mb	1	S/ 1,188.00
3	PHP 5.1.0	1	S/ 0.00
4	MySQL 5.0	1	S/ 0.00
Total		4	S/ 1,188.00

Fuente: Elaboración propia

* Se usará el servidor de la UNTELS.

3.2.2 Identificación del Scrum Master y los socios.

3.2.2.1 Entradas.

3.2.2.1.1 Propietario del producto.

Descrito en la sección 3.1.1.3.1.

3.2.2.1.2 Declaración de la visión del proyecto.

Descrito en la sección 3.1.1.3.2.

3.2.2.2 Herramienta.

3.2.2.2.1 Criterios de selección.

Para la elección del Scrum master se consideraron los siguientes criterios de selección:

1. Habilidad para la resolución de problemas, debe tener las habilidades y experiencia necesarias para ayudar a eliminar cualquier impedimento que enfrente el quipo Scrum.

2. Disponibilidad, debe estar disponible para programar, supervisar y facilitar varias reuniones.
3. Compromiso, debe estar comprometido a que el equipo Scrum esté dotado de un ambiente de trabajo propicio para asegurar la entrega exitosa del proyecto Scrum.
4. Estilo de liderazgo servicial.

En la identificación de los socios se incluyó a todos los clientes, usuarios y patrocinadores.

3.2.2.3 Salidas.

3.2.2.3.1 Scrum master identificado.

Se identificó al Scrum master:

- Scrum master: Gerard Chozo Nalvarte

3.2.2.3.2 Socios identificados.

Se identificaron a los siguientes socios:

- Estudiantes y egresados de la UNTELS.
- Personal Administrativo de la UNTELS.
- Oficina General de Tecnologías de la Información y Comunicación.
- Oficina General de Servicios Universitarios.
- Oficinas de las Carreras Profesionales.
- Empresas interesadas en practicantes universitarios.

3.2.3 Formación del equipo Scrum.

3.2.3.1 Entradas.

3.2.3.1.1 Propietario del producto.

Descrito en la sección 3.1.1.3.1.

3.2.3.1.2 Scrum Master.

Descrito en la sección 3.1.2.3.1.

3.2.3.1.3 Declaración de la visión del proyecto.

Descrito en la sección 3.1.1.3.2.

3.2.3.2 Herramienta.

3.2.3.2.1 Selección del equipo Scrum.

Para la elección del equipo Scrum se consideró lo siguiente:

- Deben contar con conocimiento en diversos campos y ser expertos en al menos uno.
- Deben ser independientes, automotivados, enfocados en el cliente con sentido de la responsabilidad y colaboración.
- Experiencia, haber trabajado en proyectos similares.
- Capacidad de tomar decisiones acertadas.
- Habilidades interpersonales.

3.2.3.3 Salida.

3.2.3.3.1 Equipo Scrum identificado.

Se identificaron a los miembros del Equipo Scrum con ayuda del propietario del producto y el Scrum master.

A continuación, se muestran los miembros del Equipo Scrum:

- Diseñador: Vladimir Peralta Nalvarte
- Desarrollador Front-end: Alexander Perez Mendoza
- Desarrollador Back-end: Alexis Honorato Laura Lopez

3.2.4 Desarrollo de épicas.

3.2.4.1 Entradas.

3.2.4.1.1 Equipo principal Scrum.

El equipo principal Scrum consiste en el equipo Scrum, el Scrum Master y el propietario del producto.

3.2.4.1.2 Declaración de la visión del proyecto.

Descrito en la sección 3.1.1.3.2.

3.2.4.2 Herramienta.

3.2.4.2.1 Reuniones del grupo de usuarios.

En la reunión del grupo de usuarios se involucró a los usuarios o clientes del producto, ellos proporcionarán información de primera mano acerca de las expectativas del usuario. Esto ayuda en la formulación de los criterios de aceptación para el producto y proporciona información valiosa para el desarrollo de épicas.

3.2.4.3 Salida.

3.2.4.3.1 Épicas.

Las épicas se redactaron cuando la mayoría de las historias de usuario son funcionalidades de alto nivel. Las épicas son historias de usuario grandes sin refinar en la lista priorizada de pendientes del proyecto, posteriormente se desglosará en historias de usuario más pequeñas que generalmente son funcionalidades simples, cortas y fáciles de implementar, que deben completarse en un sprint.

Se identificaron las siguientes épicas del proyecto:

- Análisis del proyecto.
- Diseño del prototipo.
- Maquetado del prototipo.
- Desarrollo de base de datos.
- Programación del prototipo.
- Integración.

3.2.4.3.2 Prototipos.

Los prototipos son personajes ficticios altamente detallados. Estos representarán a la mayoría de los usuarios y otros socios que pudieran no utilizar directamente el producto final. Los prototipos se crean para identificar las necesidades de los usuarios.

Creación de un prototipo:

Jaime Guardia tiene 21 años de edad y reside en Villa El Salvador. Él estudia la carrera profesional de Ingeniería de Sistemas en la Universidad Nacional Tecnológica de Lima Sur. Actualmente cursa el séptimo ciclo y ha decidido hacer sus prácticas pre

profesionales, le gustaría poder encontrar un lugar donde pueda encontrar empresas que requieran practicantes con un perfil como el suyo. También desearía poder postular a convocatorias laborales sin necesidad de ir en persona simplemente a entregar su Curriculum vitae o hacer algún trámite que fácilmente podría hacerse vía internet. Él quiere ser notificado cada vez que se realice una nueva convocatoria laboral de su interés para no tener que estar revisando constantemente si hay novedades.

3.2.5 Creación de la lista priorizada de pendientes del producto.

3.2.5.1 Entradas.

3.2.5.1.1 Equipo principal Scrum.

Descrito en la sección 3.1.4.1.1.

3.2.5.1.2 Épicas.

Descrito en la sección 3.1.4.3.1.

3.2.5.1.3 Prototipos.

Descrito en la sección 3.1.4.3.2.

3.2.5.2 Herramienta.

3.2.5.2.1 Métodos de priorización de historias de usuario.

A continuación, se presenta la técnica utilizada para dar prioridad a las historias de usuario en la lista priorizada de pendientes del producto.

Comparación por pares:

En esta técnica se prepara una lista de todas las historias de usuario en la lista priorizada de pendientes del producto. Después cada historia de usuario se toma en forma individual y se compara con otras historias en la lista, una a la vez. Cada vez que se comparan dos historias de usuario, se toma una decisión en cuanto a cuál de las dos es más importante. Por medio de este proceso, se puede generar una lista priorizada de las historias de usuario.

3.2.5.3 Salidas.

3.2.5.3.1 *Lista priorizada de pendientes del producto.*

El propietario del producto desarrolla una lista priorizada de pendientes del producto, que contiene una lista priorizada de los requerimientos del negocio y del proyecto escrito en forma de épicas, que son historias de usuario de alto nivel.

En la tabla 7 se muestra el Product Backlog del proyecto.

Tabla 7
Product Backlog del proyecto.

Código	Requerimiento	Estimación (horas)	Prioridad
RQ-01	Desarrollo de mockups	16	1
RQ-02	Modelado de la base de datos	8	1
RQ-03	Creación de la base de datos	8	1
RQ-04	Maquetado del login	4	2
RQ-05	Maquetado de la sección mi cuenta	4	2
RQ-06	Maquetado de la sección publicaciones	4	2
RQ-07	Maquetado de la sección notificaciones	4	2
RQ-08	Maquetado de la sección del editor	4	2
RQ-09	Desarrollar el módulo login	8	3
RQ-10	Desarrollar el módulo mi cuenta	8	3

RQ-11	Desarrollar el módulo publicaciones	8	3
RQ-12	Desarrollar el módulo notificaciones	8	3
RQ-13	Desarrollar el módulo editor	8	3

Fuente: Elaboración propia

3.2.5.3.2 *Criterios de terminado.*

Conjunto de reglas que se aplican a todas las historias de usuario. Una definición clara de terminado elimina la ambigüedad de los requisitos y ayuda a que el equipo se adhiera las normas de calidad obligatorias. Una historia de usuario se considera terminada cuando se le demuestra y se aprueba por el propietario del producto, que juzga sobre la base de los criterios de terminado y los criterios de aceptación de las historias de usuario.

Criterios de terminado:

- El diseño del prototipo es aprobado por el dueño del producto.
- El maquetado de las secciones es aprobado por el dueño del producto.
- El modelado y creación de base de datos es aprobado por el dueño del producto.
- La programación de las secciones es aprobada por el dueño del producto.

3.2.6 Realizar el plan de lanzamiento.

3.2.6.1 Entradas.

3.2.6.1.1 Equipo principal Scrum.

Descrito en la sección 3.1.4.1.1.

3.2.6.1.2 Socios.

Descrito en la sección 3.1.2.3.2.

3.2.6.1.3 Declaración de la visión del proyecto.

Descrito en la sección 3.1.1.3.2.

3.2.6.2 Herramienta.

3.2.6.2.1 Sesiones de planificación del lanzamiento.

La reunión de planificación del lanzamiento se llevó a cabo para desarrollar un plan de lanzamiento. El plan define cuando varios conjuntos de funcionalidad o productos utilizables será entregados al cliente. El objetivo principal de una reunión de planificación de lanzamiento es hacer que el equipo Scrum tenga una visión general de los lanzamientos y el calendario de entrega del producto que están desarrollando para que puedan alinearse con las expectativas del propietario del producto y los socios relevantes (principalmente el patrocinador del proyecto).

3.2.6.3 Salidas.

3.2.6.3.1 Cronograma de planificación del lanzamiento.

Se procedió a revisar las historias de usuario de la lista priorizada de pendientes del producto para desarrollar el cronograma de planificación del lanzamiento.

Tabla 8
Cronograma de planificación del lanzamiento

DETALLES DEL PROYECTO		
FECHA INICIO	FECHA FIN	HITO
05/03/2018	05/03/2018	Inicio del proyecto
05/03/2018	16/03/2018	Fase de análisis
19/03/2018	30/03/2018	Diseño del aplicativo
02/04/2018	04/04/2018	Modelado y creación de la base de datos
05/04/2018	06/04/2018	Diseño de los Mockups
09/04/2018	20/04/2018	Maquetación de las secciones
23/04/2018	11/05/2018	Desarrollo de los módulos
14/05/2018	25/05/2018	Integración
28/05/2018	08/06/2018	Control de calidad
11/06/2018	11/06/2018	Fin del proyecto

Fuente: Elaboración propia

3.2.6.3.2 Duración del Sprint.

Basado en los requerimientos de negocio y el cronograma de planificación del lanzamiento, el propietario del producto y el equipo Scrum se decidió que la duración del Sprint será el siguiente.

Duración del Sprint: Cuatro semanas.

3.3 Fase de Planificación y Estimación.

3.3.1 Creación de historias de usuario.

3.3.1.1 Entradas.

3.3.1.1.1 Equipo principal Scrum.

Descrito en la sección 3.1.4.1.1.

3.3.1.1.2 Lista priorizada de pendientes del producto.

Descrito en la sección 3.1.5.3.1.

3.3.1.1.3 Criterios de terminado.

Descrito en la sección 3.1.5.3.2.

3.3.1.1.4 Prototipos.

Descrito en la sección 3.1.4.3.2.

3.3.1.2 Herramienta.

3.3.1.2.1 Experiencia en la redacción de historias de usuario.

El propietario del producto gracias a su interacción con los socios, su conocimiento del negocio, su experiencia y las aportaciones del equipo, desarrolla las historias de usuario que formaran la lista inicial de pendientes del producto para el proyecto.

3.3.1.3 Salidas.

3.3.1.3.1 Historias de usuario.

Las historias de usuario son una forma simple de documentar los requerimientos y funcionalidades que desea el usuario final. Los requerimientos de las historias de usuario son oraciones breves, sencillas y fáciles de entender. A continuación, se listan las historias de usuario.

Historias de usuario:

- Como cliente quiero ver un bosquejo inicial de las interfaces con las que los usuarios interactuarán dentro del aplicativo.
- Como cliente quiero que la información de los usuarios, empresas, publicaciones y notificaciones estén disponibles para poder usarlas cuando lo desee.
- Como cliente quiero que el sistema permita verificar la identidad de un usuario para que pueda ingresar al aplicativo.
- Como cliente quiero que después de iniciar sesión el usuario pueda navegar entre las diferentes secciones de forma fácil y rápida.
- Como cliente quiero que el sistema permita que un trabajador pueda cumplir el rol de editor, el cual podrá crear, editar y eliminar publicaciones en el aplicativo.
- Como usuario quiero que el prototipo cuente con una sección “mi perfil” en la que pueda visualizar mis datos, notificaciones, etc.

- Como cliente quiero que los usuarios puedan acceder a la sección “Publicaciones” para poder visualizar las convocatorias laborales publicadas y poder filtrarlas mediante un buscador.
- Como cliente quiero que los usuarios puedan dar clic a una publicación para poder ingresar a la publicación completa y detallada.
- Como cliente quiero que los usuarios puedan acceder a una sección de “notificaciones”, donde visualicen las nuevas publicaciones de convocatorias laborales según el orden de publicación.
- Como cliente quiero que el sistema envíe notificaciones a los usuarios al momento de publicar una nueva convocatoria laboral o al actualizar una existente.

3.3.1.3.2 Criterios de aceptación de las historias de usuario.

Los criterios de aceptación brindan la objetividad requerida para que las historias de usuario se consideren terminadas o no terminadas durante la revisión del sprint. El propietario del producto define y comunica los criterios de aceptación al equipo Scrum. Cada historia de usuario cuenta con su respectivo criterio de aceptación. A continuación, se listan los criterios de aceptación para las historias de usuario.

Criterios de aceptación:

- Imágenes de cada interfaz a modo de mockups.
- Es posible acceder a la información y usarla.
- Iniciar sesión al digitar correctamente el usuario y la contraseña.
- Navegar fácilmente entre las diferentes secciones.
- El editor inicia sesión y puede insertar, editar, eliminar publicaciones.
- Ingresar a la sección “mi perfil” y poder visualizar la información correspondiente.
- Ingresar a la sección “Publicaciones”, poder visualizar la información correspondiente y poder buscar publicaciones mediante un buscador por palabras.
- Dar clic a una publicación e ingresar al detalle de la publicación donde visualice la información correspondiente.
- Ingresar a la sección “notificaciones”, se debe poder visualizar la información correspondiente por orden de publicación.
- Recibir una notificación luego que se realice una nueva publicación o se actualice una existente.

3.3.2 Aprobación, estimación y asignación de historias de usuario.

3.3.2.1 Entradas.

3.3.2.1.1 Equipo principal Scrum.

Descrito en la sección 3.1.4.1.1.

3.3.2.1.2 *Historias de usuario.*

Descrito en la sección 3.2.1.3.1.

3.3.2.1.3 *Criterios de aceptación de las historias de usuario.*

Descrito en la sección 3.2.1.3.2.

3.3.2.2 **Herramienta.**

3.3.2.2.1 *Reuniones del grupo de usuarios.*

Descrito en la sección 3.1.4.2.1.

3.3.2.3 **Salidas.**

3.3.2.3.1 *Historias de usuario aprobadas, estimadas y asignadas.*

Las historias de usuario que son entradas en este proceso tienen estimaciones de alto nivel, estas estimaciones las implementa el propietario del producto para aprobar las historias de usuario para el sprint. Una vez aprobadas, las historias de usuario son estimadas por el equipo. A continuación, se listan las historias de usuario aprobadas, estimadas y asignadas.

Tabla 9
Historia de usuario 1

HISTORIA DE USUARIO 1	MOCKUPS
Como cliente quiero ver un bosquejo inicial de las interfaces con las que los usuarios interactuarán dentro del aplicativo.	
Prioridad: 1	Estimación: 8 h
Criterios de aceptación: Imágenes de cada interfaz a modo de mockups.	

Fuente: elaboración propia

Tabla 10
Historia de usuario 2

HISTORIA DE USUARIO 2	BASE DE DATOS
Como cliente quiero que la información de los usuarios, empresas, publicaciones y notificaciones estén disponibles para poder usarlas cuando lo desee.	
Prioridad: 1	Estimación: 16 h
Criterios de aceptación: Es posible acceder a la información y usarla.	

Fuente: elaboración propia

Tabla 11
Historia de usuario 3

HISTORIA DE USUARIO 3	LOGIN DE USUARIO
Como cliente quiero que el sistema permita verificar la identidad de un usuario para que pueda ingresar al aplicativo.	
Prioridad: 3	Estimación: 8 h
Criterios de aceptación: Iniciar sesión al digitar correctamente el usuario y la contraseña.	

Fuente: elaboración propia

Tabla 12
Historia de usuario 4

HISTORIA DE USUARIO 4	TABS
Como cliente quiero que después de iniciar sesión el usuario pueda navegar entre las diferentes secciones de forma fácil y rápida.	
Prioridad: 3	Estimación: 4 h
Criterios de aceptación: Navegar fácilmente entre las diferentes secciones.	

Fuente: elaboración propia

Tabla 13
Historia de usuario 5

HISTORIA DE USUARIO 5	PUBLICACIÓN DE CONVOCATORIAS LABORALES
Como cliente quiero que el sistema permita que un trabajador pueda cumplir el rol de editor, el cual podrá crear, editar y eliminar publicaciones en el aplicativo.	
Prioridad: 3	Estimación: 8 h
Criterios de aceptación: El editor inicia sesión y puede insertar, editar, eliminar publicaciones.	

Fuente: elaboración propia

Tabla 14
Historia de usuario 6

HISTORIA DE USUARIO 6	SECCIÓN MI CUENTA
Como usuario quiero que el prototipo cuente con una sección “mi perfil” en la que pueda visualizar mis datos, notificaciones, etc.	
Prioridad: 2	Estimación: 4 h
Criterios de aceptación: Ingresar a la sección “mi perfil” y poder visualizar la información correspondiente.	

Fuente: elaboración propia

Tabla 15
Historia de usuario 7

HISTORIA DE USUARIO 7	SECCIÓN PUBLICACIONES
Como cliente quiero que los usuarios puedan acceder a la sección “Publicaciones” para poder visualizar las convocatorias laborales publicadas y poder filtrarlas mediante un buscador.	
Prioridad: 2	Estimación: 6 h
Criterios de aceptación: Ingresar a la sección “Publicaciones”, poder visualizar la información correspondiente y poder buscar publicaciones mediante un buscador por palabras.	

Fuente: elaboración propia

Tabla 16
Historia de usuario 8

HISTORIA DE USUARIO 8	DETALLE PUBLICACIÓN
Como cliente quiero que los usuarios puedan dar clic a una publicación para poder ingresar a la publicación completa y detallada.	
Prioridad: 3	Estimación: 8 h
Criterios de aceptación: Dar clic a una publicación e ingresar al detalle de la publicación donde visualice la información correspondiente.	

Fuente: elaboración propia

Tabla 17
Historia de usuario 9

HISTORIA DE USUARIO 9	SECCIÓN NOTIFICACIONES
Como cliente quiero que los usuarios puedan acceder a una sección de “notificaciones”, donde visualicen las nuevas publicaciones de convocatorias laborales según el orden de publicación.	
Prioridad: 3	Estimación: 8 h
Criterios de aceptación: Ingresar a la sección “notificaciones”, se debe poder visualizar la información correspondiente por orden de publicación.	

Fuente: elaboración propia

Tabla 18
Historia de usuario 10

HISTORIA DE USUARIO 10	ENVIAR NOTIFICACIONES
Como cliente quiero que el sistema envíe notificaciones a los usuarios al momento de publicar una nueva convocatoria laboral o al actualizar una existente.	
Prioridad: 3	Estimación: 8 h
Criterios de aceptación: Recibir una notificación luego que se realice una nueva publicación o se actualice una existente.	

Fuente: elaboración propia

3.3.3 Creación de la lista de pendientes del Sprint.

3.3.3.1 Entradas.

3.3.3.1.1 Equipo principal Scrum.

Descrito en la sección 3.1.4.1.1.

3.3.3.1.2 Duración del Sprint.

Descrito en la sección 3.1.6.3.2.

3.3.3.2 Herramienta.

3.3.3.2.1 Reuniones de planificación del Sprint.

Durante la reunión de planificación del Sprint, las historias de usuario se someten a discusión del equipo Scrum. El equipo Scrum elabora la lista de pendientes del Sprint, con el uso de historias de usuario.

3.3.3.3 Salidas.

3.3.3.3.1 Lista de pendientes del Sprint.

La lista de pendientes del Sprint, es la lista de las tareas que ejecutará el equipo Scrum en el siguiente Sprint. En la tabla 18 se presentan los Sprints Backlog de las 3 iteraciones que forman parte del desarrollo del producto.

Tabla 19
Especificación de los Sprint Backlog

Iteración	Código	Requerimiento
Sprint 1	RQ-01	Desarrollo de mockups
	RQ-02	Modelado de la base de datos
	RQ-03	Creación de la base de datos
Sprint 2	RQ-04	Maquetado del login
	RQ-05	Maquetado de la sección mi cuenta
	RQ-06	Maquetado de la sección publicaciones
	RQ-07	Maquetado de la sección notificaciones
	RQ-08	Maquetado de la sección del editor
Sprint 3	RQ-09	Desarrollar el módulo login
	RQ-10	Desarrollar el módulo mi cuenta
	RQ-11	Desarrollar el módulo publicaciones
	RQ-12	Desarrollar el módulo notificaciones
	RQ-13	Desarrollar el módulo editor

Fuente: Elaboración propia

3.4 Fase de Implementación.

3.4.1 Creación de entregables.

3.4.1.1 Entradas.

3.4.1.1.1 *Equipo principal Scrum.*

Descrito en la sección 3.1.4.1.1.

3.4.1.1.2 *Lista de pendientes del Sprint.*

Descrito en la sección 3.1.5.3.1.

3.4.1.2 Herramienta.

3.4.1.2.1 *Experiencia del equipo.*

La experiencia colectiva de los miembros del equipo Scrum es necesaria para entender las historias de usuario y las tareas en la lista de pendientes del Sprint a fin de crear los entregables finales. El juicio y la experiencia se aplican en todos los aspectos técnicos y

administrativos del proyecto durante el proceso de creación de entregables.

3.4.1.3 Salidas.

3.4.1.3.1 *Entregables del Sprint.*

Al final de cada Sprint se completó un incremento de producto o entregable. El entregable incluye todas las características y funcionalidades definidas en las historias de usuario que se incluyen en el Sprint y fueron evaluadas satisfactoriamente.

Realización del Sprint 1.

Desarrollo de mockups.

Como primer paso en la construcción del prototipo se procede a crear los mockups, los que mostrarán a un alto nivel las interfaces con las que interactuará el usuario. Se emplearon técnicas de interfaz de usuario para asegurar que el prototipo fuera agradable al usuario y a la vez funcional.

A continuación, se listan los mockups realizados con la herramienta Balsamiq.



Figura 2. Mockup login
Fuente: Elaboración propia

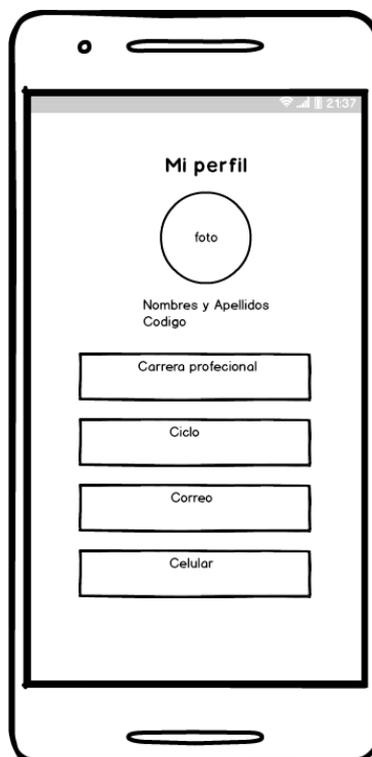


Figura 3. Mockup mi perfil
Fuente: Elaboración propia

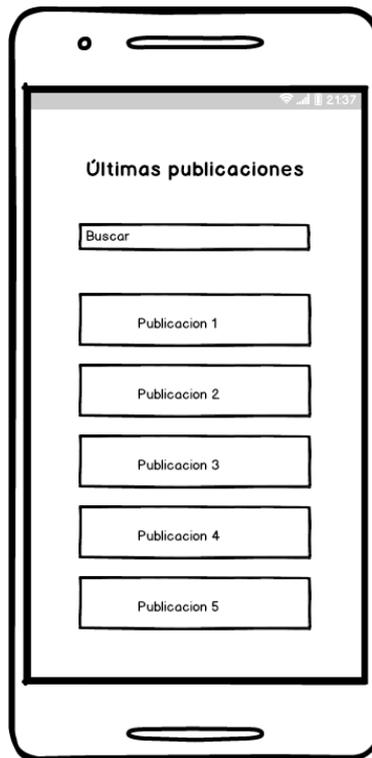


Figura 4. Mockup publicaciones
Fuente: Elaboración propia

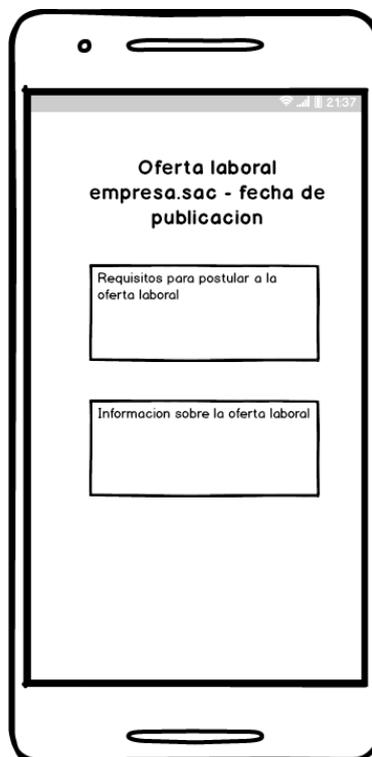


Figura 5. Mockup detalle publicación
Fuente: Elaboración propia

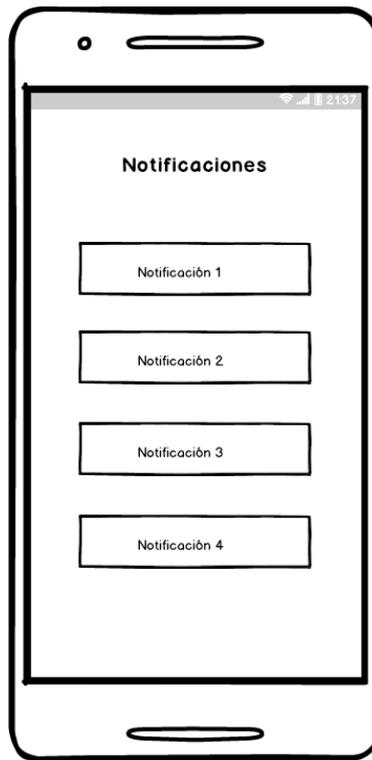


Figura 6. Mockup notificaciones
Fuente: Elaboración propia

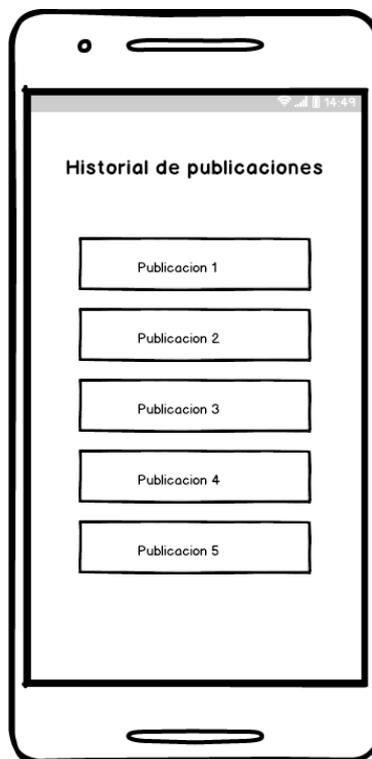


Figura 7. Mockup historial publicador
Fuente: Elaboración propia

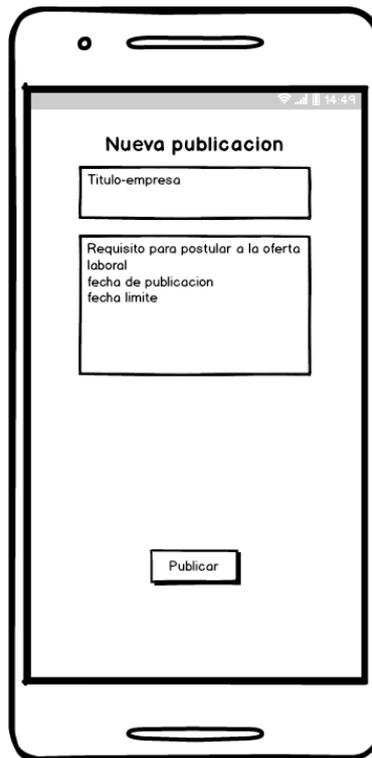


Figura 8. Mockup nueva publicación
Fuente: Elaboración propia

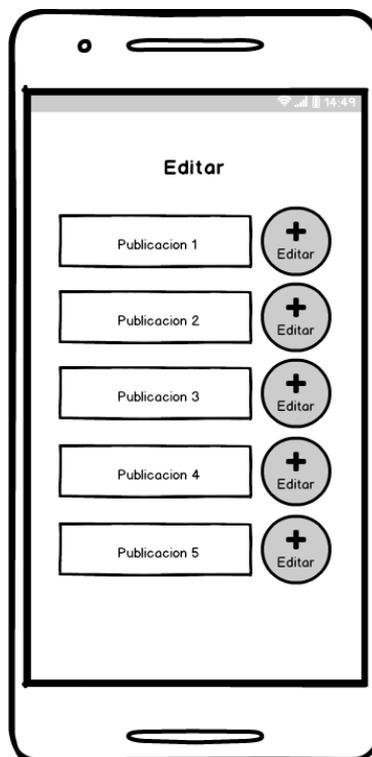


Figura 9. Mockup editar publicación
Fuente: Elaboración propia

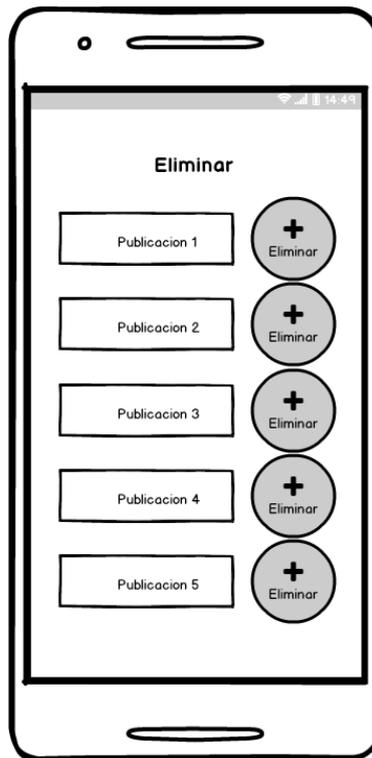


Figura 10. Mockup eliminar publicación
Fuente: Elaboración propia

Modelado y creación de la Base de Datos.

Para el desarrollo del prototipo es necesario contar con una fuente de datos, como lo indica el requerimiento RQ-02, que permita al usuario de la aplicación verificar su identidad, mediante la validación de su “usuario” y “contraseña”, para poder usar el aplicativo. En la figura 11 se presenta el modelo de base de datos MySQL con el que trabajará el servidor web, para realizar este requerimiento fue necesario el uso del software MySQL Workbench (MySQL, 2015).

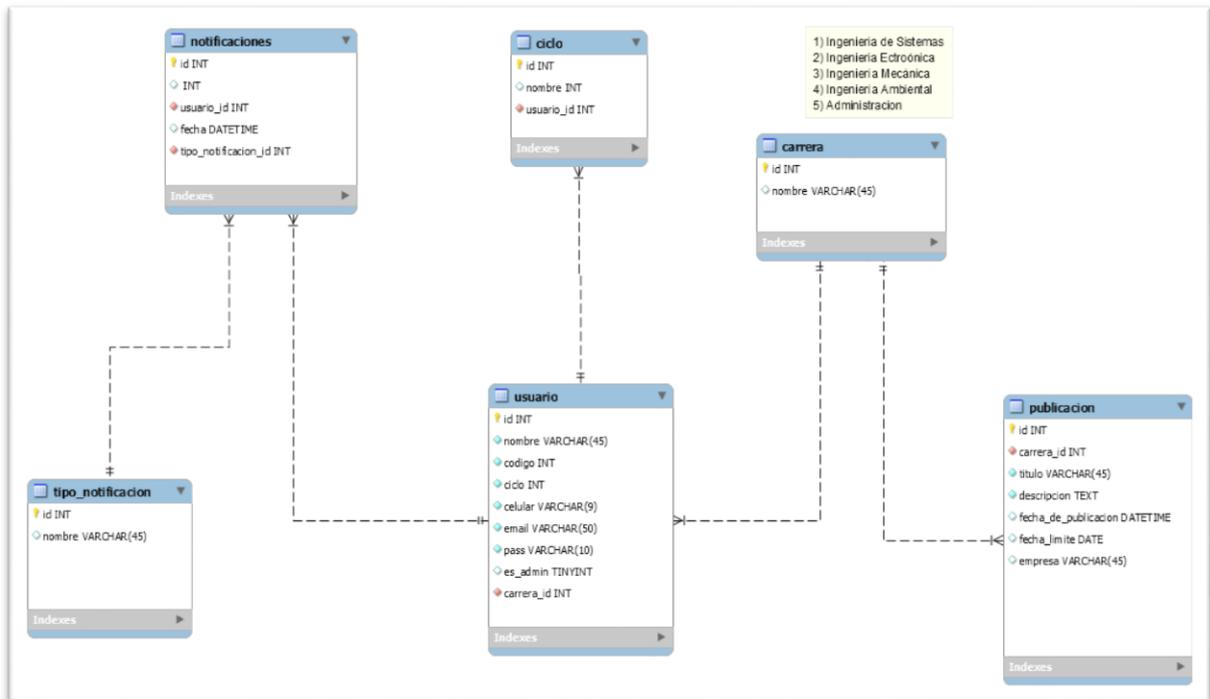


Figura 11. Modelo de base de datos
Fuente: Elaboración propia

A partir del modelo de base de datos se genera y ejecuta el script SQL necesario que permita su creación en el SGBD MySQL (RQ-03).

Realización del Sprint 2.

Después de elaborados los mockups se procede al desarrollo front-end, para asegurar que el prototipo cumpla con las expectativas del usuario, se emplearon técnicas de experiencia de usuario. Para tener un entorno de trabajo adecuado fue necesario preparar las herramientas a usar tal como se describe a continuación.

Creación del repositorio.

Primero se instala el control de versiones GIT (v2.16.2), luego se crea el repositorio en el servicio de alojamiento Bitbucket con el nombre “Bolsa de trabajo UNTELS” donde subiremos nuestras iteraciones a la nube.

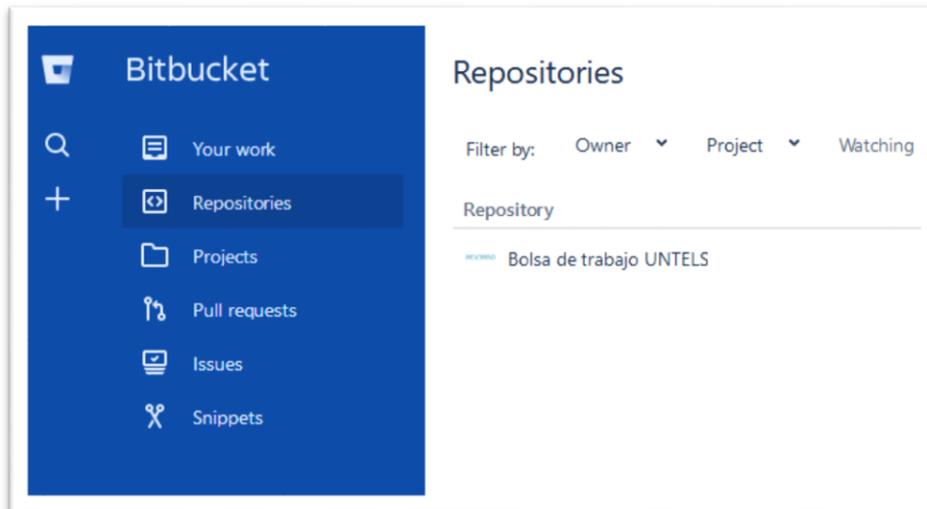


Figura 12. Creación del repositorio del proyecto
Fuente: Elaboración propia

Creación del proyecto *aplicativo* en Ionic

Siguiendo con la documentación oficial de Ionic:

- Instalación de NodeJS. Necesario para usar las funcionalidades de Ionic.
- Instalación de Cordova. Una vez instalado node.js digitar en la consola: `npm install -g cordova@ionic`.
- Creación del proyecto. Ejecutar el siguiente comando de creación `ionic start tagger-mobiletabs`. Con lo cual el proyecto contará con un estilo de sidebar, el cual se ajusta mejor a los propósitos del proyecto.

Subir el proyecto al repositorio.

Luego de tener el proyecto debidamente instalado se procede a subirlo al repositorio. Para esto debemos iniciar GIT en nuestro proyecto, hecho esto, ya será posible subirlo al repositorio y todo estará listo para empezar a trabajar, llevando un control de versiones de manera segura gracias a GIT.

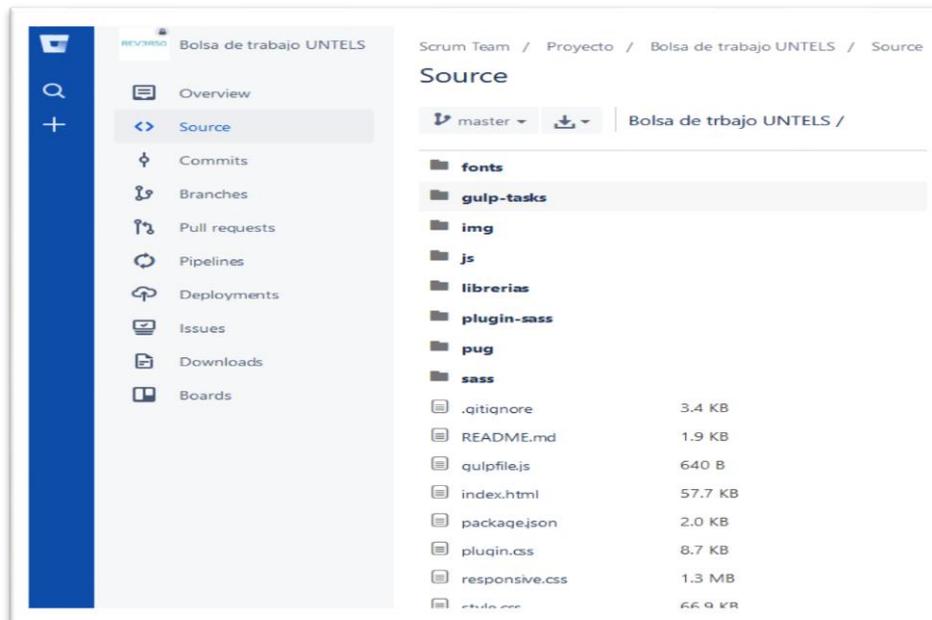


Figura 13. Proyecto subido a Bitbucket
Fuente: Elaboración propia

A continuación, se muestran los issues del proyecto

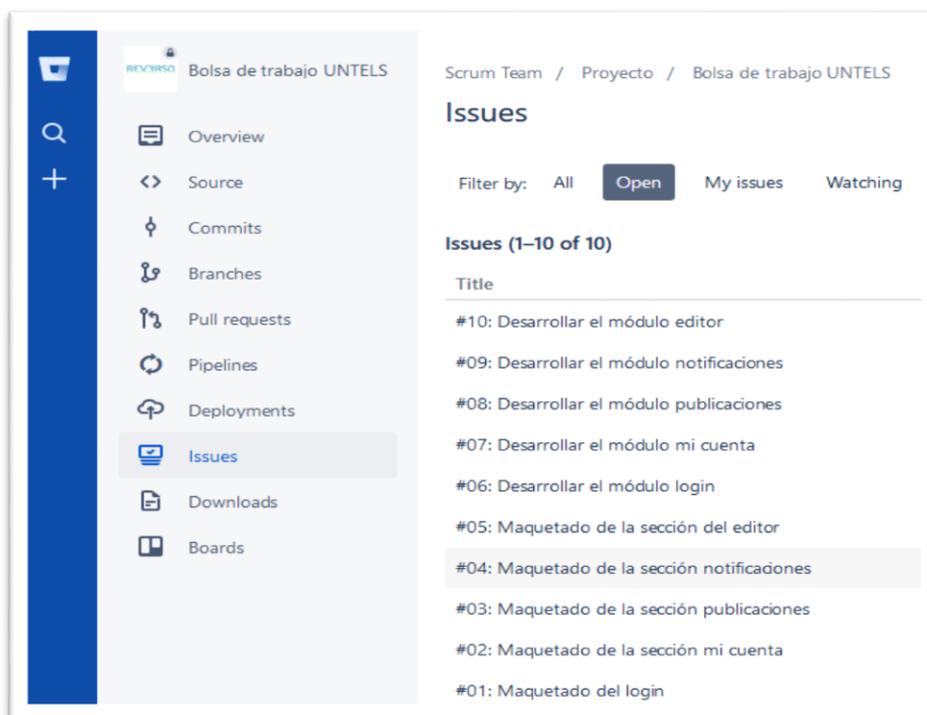


Figura 14. Issues del proyecto en Bitbucket
Fuente: Elaboración propia

Maquetado de los mockups.

Es importante mencionar que al ser un prototipo desarrollado en Ionic, el cual ya viene integrado con AngularJS, estamos facultados de aprovechar la versatilidad y propiedades de la tecnología web. Ionic, por su parte, incorpora el uso de estilos Css predefinidos, colores, elementos html, funcionalidades JavaScript y más, todo ello conservando un estilo responsive para la adaptación al móvil y para lo cual define sus propias reglas de ayuda.

El desarrollo del proyecto creado en Ionic, tiene el siguiente orden.

El proyecto desarrollo, tiene el siguiente orden de ejecución.

Armado del login.

Corresponde a la portada de la aplicación en la que el usuario deberá ingresar su “usuario” y “contraseña” para poder iniciar sesión.



Figura 15. Maquetación login
Fuente: Elaboración propia

Armado de la sección “Mi Perfil”.

El perfil personal es la sección donde se visualizará la información personal del usuario.



Figura 16. Maquetación mi perfil
Fuente: Elaboración propia

Armado de la sección de “Publicaciones”.

Las publicaciones de las convocatorias laborales son una de las partes más importantes en la elaboración del proyecto.



Figura 17. Maquetación publicaciones
Fuente: Elaboración propia



Figura 18. Maquetación detalle publicación
Fuente: Elaboración propia

Armado de la sección de Notificaciones.

En esta sección del aplicativo el usuario podrá visualizar las notificaciones recibidas, así podrá mantenerse informado sobre las nuevas publicaciones de convocatorias laborales.



Figura 19. Maquetación notificaciones
Fuente: Elaboración propia

Armado de la sección del editor.

El aplicativo contempla el rol de un editor, el cual será el encargado de la gestión de la bolsa de trabajo, como el insertar una nueva publicación, editar publicaciones y eliminar publicaciones.



Figura 20. Maquetación historial publicador
Fuente: Elaboración propia



Figura 21. Maquetación nueva publicación
Fuente: Elaboración propia

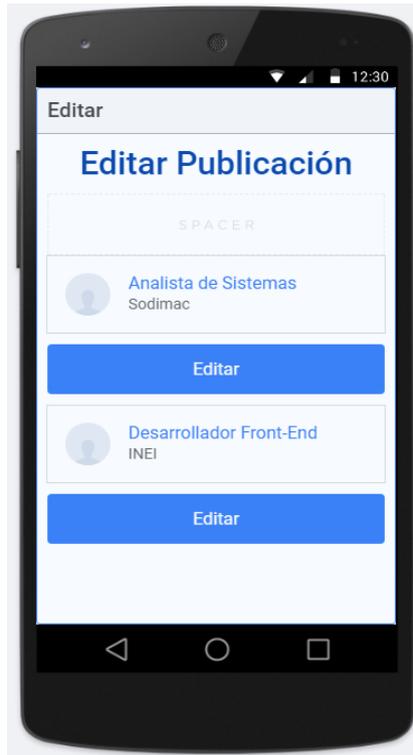


Figura 22. Maquetación editar publicación
Fuente: Elaboración propia

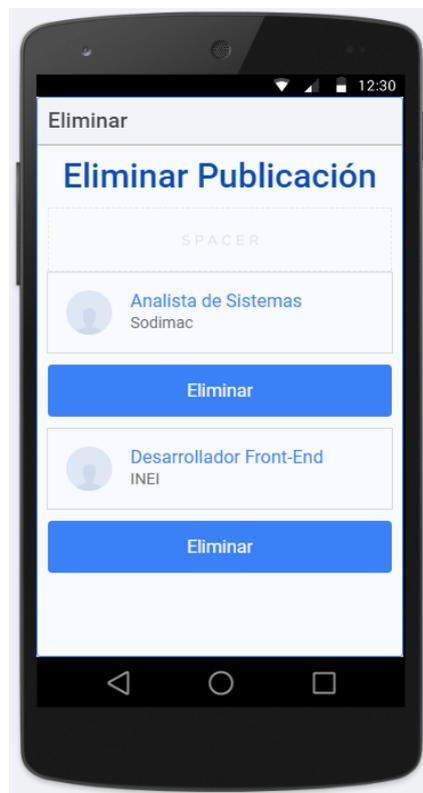


Figura 23 Maquetación eliminar publicación
Fuente: Elaboración propia

Una vez realizada la maquetación de las secciones, el árbol de iteraciones dentro de Bitbucket quedaría de la siguiente manera:

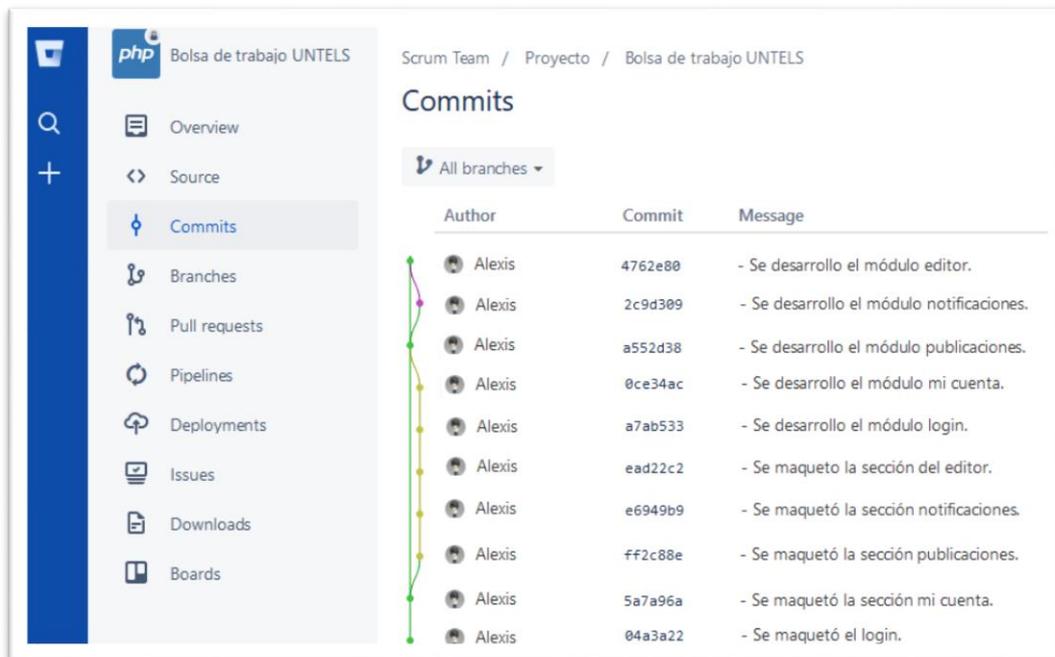


Figura 24. Commits del proyecto en Bitbucket
Fuente: Elaboración propia

Realización del Sprint 3.

Luego de terminar la maquetación de todas las secciones se procede al desarrollo back-end. El desarrollo de los módulos se realizará con php y con el IDE de programación PhpStorm.

```
database.php x
1 <?php
2 $server = 'localhost';
3 $username = 'root';
4 $password = 'root';
5 $database = 'bolsa_de_trabajo';
6
7 try{
8     $conn = new PDO( dsn: "mysql:host=$server;dbname=$database;", $username, $password);
9 } catch(PDOException $e){
10     die( "Connection failed: " . $e->getMessage());
11 }
```

Figura 25. Conexión de base de datos
Fuente: Elaboración propia

```

1 <?php
2
3 session_start();
4
5 require 'database.php';
6
7 if( isset($_SESSION['user_id']) ){
8
9     $records = $conn->prepare('SELECT id,email,password FROM users WHERE id = :id');
10    $records->bindParam(':id', $_SESSION['user_id']);
11    $records->execute();
12    $results = $records->fetch(PDO::FETCH_ASSOC);
13
14    $user = NULL;
15
16    if( count($results) > 0){
17        $user = $results;
18    }
19
20 }
21
22
23
24 <!DOCTYPE html>
25 <html>
26 <head>
27 <title>Welcome to your Web App</title>
28 <link rel="stylesheet" type="text/css" href="assets/css/style.css">
29 <link href="http://fonts.googleapis.com/css?family=Comfortaa" rel="stylesheet" type="text/css">
30 </head>
31 <body>
32
33 <div class="header">
34 <a href="/">Your App Name</a>
35 </div>
36
37 <?php if( !empty($user) ): ?>
38
39 <br />Welcome <?= $user['email']; ?>
40 <br /><br />You are successfully logged in!
41 <br /><br />
42 <a href="logout.php">Logout</a>
43
44 <?php else: ?>
45
46 <h1>Please Login or Register</h1>
47 <a href="login.php">Login</a> or
48 <a href="register.php">Register</a>
49
50 </body>
51 </html>

```

Figura 26. Inicio de sesión
Fuente: Elaboración propia

```

1
2
3
4 if( isset($_SESSION['user_id']) ){
5     header( string: "Location: /" );
6 }
7
8
9 require 'database.php';
10
11 if(!empty($_POST['email']) && !empty($_POST['password'])){
12
13     $records = $conn->prepare('SELECT id,email,password FROM users WHERE email = :email');
14     $records->bindParam(':email', $_POST['email']);
15     $records->execute();
16     $results = $records->fetch(PDO::FETCH_ASSOC);
17
18     $message = '';
19
20     if(count($results) > 0 && password_verify($_POST['password'], $results['password'])){
21
22         $_SESSION['user_id'] = $results['id'];
23         header( string: "Location: /" );
24
25     } else {
26         $message = 'Sorry, those credentials do not match!';
27     }
28
29 }
30
31
32
33 <!DOCTYPE html>
34 <html>
35 <head>
36 <title>Login Below</title>
37 <link rel="stylesheet" type="text/css" href="assets/css/style.css">
38 <link href="http://fonts.googleapis.com/css?family=Comfortaa" rel="stylesheet" type="text/css">
39 </head>
40 <body>
41
42 <div class="header">
43 <a href="/">Your App Name</a>
44 </div>
45
46 <?php if( empty($message) ): ?>
47 <p><?= $message ?></p>
48 <?php endif: ?>
49
50 <h1>Login</h1>
51 <span>or <a href="register.php">register here</a></span>
52
53 <form action="login.php" method="POST">

```

Figura 27. Modulo Login
Fuente: Elaboración propia

3.5 Fase de Revisión y Retrospectiva.

3.5.1 Demostración y validación del sprint.

3.5.1.1 Entradas.

3.5.1.1.1 Equipo principal Scrum.

Descrito en la sección 3.1.4.1.1.

3.5.1.1.2 Entregables del Sprint.

Descrito en la sección 3.3.1.1.3.

3.5.1.1.3 Lista de pendientes del Sprint.

Descrito en la sección 3.2.3.3.1.

3.5.1.1.4 Criterios de terminado.

Descrito en la sección 3.1.5.3.2.

3.5.1.1.5 Criterios de aceptación de las historias de usuario.

Descrito en la sección 3.2.1.3.2.

3.5.2 Herramienta.

3.5.2.1 Reunión de revisión del Sprint.

Los miembros del equipo principal de Scrum y los socios relevantes participan de la reunión de revisión del Sprint para aceptar los entregables que cumplan con los criterios de aceptación de las historias de usuario. Estas reuniones se convocan al final de cada Sprint.

3.5.2.2 Salidas.

3.5.2.2.1 *Entregables aceptados.*

Los entregables que cumplen con los criterios de aceptación de las historias de usuario son aceptados por el propietario del producto.

3.5.3 Retrospectiva de Sprint.

3.5.3.1 Entradas.

3.5.3.1.1 *Scrum Master.*

Descrito en la sección 3.1.2.3.1.

3.5.3.1.2 *Equipo Scrum.*

Descrito en la sección 3.1.3.3.1.

3.5.3.2 Herramienta.

3.5.3.2.1 *Reunión de retrospectiva del Sprint.*

Todos los miembros del equipo Scrum asisten a la reunión la cual es organizada y moderada por el Scrum Master, también asiste el propietario del producto, un integrante del equipo documenta las discusiones y los elementos para acciones futuras. La discusión en la reunión abarca tanto lo que salió bien como lo que salió mal. Los objetivos primordiales de esta reunión son identificar tres elementos:

1. Las cosas que el equipo necesita seguir haciendo: Mejores prácticas.

2. Las cosas que el equipo necesita empezar a hacer: Mejoras en el proceso.
3. Las cosas que el equipo necesita dejar de hacer: Problemas de proceso y embotellamiento.

3.5.3.3 Salidas.

3.5.3.3.1 Mejoras accionables aceptadas.

Las mejoras accionables aceptadas son los resultados primarios del proceso de retrospectiva del Sprint. Forman parte de la lista de elementos accionables que ha elaborado el equipo para hacer frente a los problemas y mejorar los procesos a fin de mejorar también su desempeño en futuros Sprints.

En este proceso el Scrum Master y el equipo Scrum se reunieron para discutir las lecciones aprendidas durante el sprint. Dicha información se documenta como lecciones aprendidas, las cuales se puedan implementar en futuros Sprints.

3.6 Fase de Lanzamiento.

3.6.1 Envío de entregables.

3.6.1.1 Entradas.

3.6.1.1.1 Propietario del producto.

Descrito en la sección 3.1.1.3.1.

3.6.1.1.2 Socios.

Descrito en la sección 3.1.2.3.2.

3.6.1.1.3 *Entregables aceptados.*

Descrito en la sección 3.4.1.3.1.

3.6.1.1.4 *Cronograma de planificación del lanzamiento.*

Descrito en la sección 3.1.6.3.1.

3.6.1.2 Herramienta.

3.6.1.2.1 *Métodos de desplazamiento organizacional.*

Dependiendo del producto a entregarse, el desplazamiento puede ser remoto o puede incluir el envío físico o la transición de un artículo.

3.6.1.3 Salidas.

3.6.1.3.1 *Acuerdo de entregables funcionales.*

Los entregables que cumplen con los criterios de aceptación, reciben el cierre formal del negocio y la aprobación formal por parte del cliente o del patrocinador. Obtener la aceptación formal del cliente es fundamental para el reconocimiento de los ingresos.

En este proceso se procedió a hacer la entrega de los entregables aceptados a los socios relevantes.

3.6.2 Retrospectiva del proyecto.

3.6.2.1 Entradas.

3.6.2.1.1 *Equipo principal de Scrum.*

Descrito en la sección 3.1.4.1.1.

3.6.2.2 Herramienta.

3.6.2.2.1 Reunión de retrospectiva del Sprint.

La reunión de retrospectiva del Sprint es una reunión para determinar las formas en las que la colaboración y eficacia del equipo puede mejorarse en futuros proyectos. También se analizan las oportunidades positivas, negativas y potenciales para mejorar. Entre los asistentes se encuentran el equipo del proyecto, propietario del producto y los socios. Durante la reunión se documentan las lecciones aprendidas.

3.6.2.3 Salidas.

3.6.2.3.1 Mejoras accionables aceptadas.

3.6.2.3.2 Elementos de acción asignados y fechas límite.

Una vez que se han elaborado y refinado los elementos de acción asignados y las fechas límite, el equipo Scrum puede considerar los puntos de acción para implementar las mejoras. Cada elemento de acción contará con una fecha límite de conclusión.

En este proceso se concluyó el proyecto, los socios y los miembros del equipo principal de Scrum se reunieron para hacer una retrospectiva del proyecto, identificar, documentar e internalizar las lecciones que se aprendieron.

CONCLUSIONES

Gracias a las características que posee una aplicación web progresiva, se logró desarrollar un prototipo de bolsa de trabajo, el cual permitió automatizar los procesos para la gestión de prácticas pre profesionales y seguimiento al egresado. Con esto se redujo el tiempo empleado para dichos procesos, de cinco minutos a un minuto.

Scrum a través de sus herramientas como la reunión de la visión del proyecto, en la que participaron el propietario del producto y los socios, se logró identificar los requerimientos para el desarrollo del proyecto y poder cumplir con las expectativas de los socios. Así mismo por medio de las historias de usuario se logró documentar de forma simple, los requerimientos y funcionalidades que el usuario final deseaba.

El uso de la metodología Scrum como marco de trabajo facilitó que los requerimientos sean atendidos de manera rápida y ajustable. Así mismo permitió gestionar el desarrollo del proyecto con un enfoque iterativo e incremental, siguiendo las fases definidas y procesos bien estructurados. Los roles, eventos y artefactos de Scrum, al estar bien detallados y ser fáciles de usar, permitieron maximizar el valor del producto, a la vez que se minimizaron los recursos empleados.

Las aplicaciones web progresivas al combinar lo mejor de la web tradicional y las aplicaciones móviles nativas, permitieron que el prototipo funcione para todos los usuarios, sin importar el dispositivo, sistema operativo o navegador usado, además de un tiempo de desarrollo menor. Gracias al Shell app se logró navegación, interacción y apariencia muy similar a las aplicaciones móviles nativas. También se logró navegar con redes débiles, sin conexión y recibir notificaciones push, para una mejor experiencia de usuario, esto gracias al Service Worker.

RECOMENDACIONES

Se recomienda integrar el prototipo desarrollado con los sistemas que posee la UNTELS y promover su uso entre los estudiantes y egresados.

Diseñar e implementar un protocolo de seguridad que complemente la gestión de riesgos y capacitar al personal que se encargará de administrar y gestionar el prototipo de bolsa de trabajo.

A partir de este prototipo se puede desarrollar un módulo enfocado en controlar y monitorear, tanto la situación laboral como académica de los alumnos egresados. También es posible desarrollar aplicaciones web progresivas para solucionar los problemas que afectan al comedor universitario, la biblioteca, el gimnasio, entre otros.

Es imprescindible que las autoridades de la UNTELS se comprometan en asegurar el constante funcionamiento del aplicativo. Así mismo se debe incluir dentro del presupuesto anual, la asignar de recursos económicos para el mantenimiento y actualización del aplicativo.

BIBLIOGRAFÍA

- Aguirre Chacón, L. T., & Sinche Ricra, H. J. (2013). *Diseño de una aplicación móvil para la consulta académica de la FIIS-UTP*. Obtenido de http://pis1.wikispaces.com/file/view/Presentacion+Final_Tesis+I.pdf
- Alania Mamani, J. W., & Apaza Lizarraga, N. E. (2014). *Aplicación web de bolsa de empleo de empresas y personas utilizando el api de google maps para la accesibilidad a la información de oportunidades laborales en la región de Puno, 2013*. Obtenido de <http://repositorio.unap.edu.pe/handle/UNAP/1675>
- Alegsa. (2010). *¿Qué son los prototipos de un sistema informático?* Obtenido de <http://www.alegsa.com.ar/Diccionario/C/7664.php>
- Alegsa. (2015). *Definición de smartphone*. Obtenido de <http://www.alegsa.com.ar/Dic/smartphone.php>
- Arquitectura de las aplicaciones Web. (2015). *Arquitectura de las aplicaciones Web* Obtenido de <https://studylib.es/doc/750605/>
- Casillas Santillán, L. A., Gibert Ginesta, M., & Pérez Mora, Ó. (2008). *Bases de datos en MySQL*. Obtenido de http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06_M2109_02151.pdf
- Caudevilla, G. (2010). *Metodología Ágil: SCRUM*. Obtenido de <http://developing.frogtek.org/wp-content/uploads/2010/07/Imagen-122.png>
- Caudevilla, G. (2010). *Waterfall vs SCRUM vs Kanban (II)*. Obtenido de <http://developing.frogtek.org/tag/daily-scrum/>
- Delía, L., Galdamez, N., Thomas, P., & Pesado, P. (2013). *Un análisis experimental de tipo de aplicaciones para dispositivos móviles*. Obtenido de

https://digital.cic.gba.gob.ar/bitstream/handle/11746/2091/Documento_completo.pdf-PDFA.pdf?sequence=1

Departamento de Informática de la Universidad de Valladolid. (2006). *El modelo Cliente/Servidor*. Obtenido de http://www.infor.uva.es/~fdiaz/sd/2005_06/doc/SD_TE02_20060305.pdf

Empresa, (2017). *Empresa*. Obtenido de <https://es.wikiquote.org/wiki/Empresa>

Espinoza Díaz, M. L., & Pérez Cuestas, J. A. (2009). *Sistema Web de "Bolsa de Empleo" para el Ministerio de Trabajo y Empleo del Ecuador*. Obtenido de <http://repositorio.espe.edu.ec/handle/21000/730>

Gómez Garduza, G. B. (2010). *Sistema de información para bolsa de trabajo*. Obtenido de <https://es.scribd.com/document/229315564/Gabriela-Gomez>

González, D., & Marcos Mora, M. C. (2013). *Responsive web desing: diseño multidispositivo para mejorar la experiencia de usuario*. Obtenido de <http://bid.ub.edu/pdf/31/es/gonzalez2.pdf>

Herramientas de búsqueda de empleo, (2013). *Herramientas de busqueda de empleo*. Obtenido de <https://lanmerkatua.wordpress.com/2013/03/07/herramientas-de-busqueda-de-empleo/>

Jiménez Ortega, R. (2017). *Curso de HTML5 desde cero*. Obtenido de <https://www.pdf-manual.es/programacion-web/html/133-curso-de-html5-desde-cero.html>

Laura, A. (2018). *Sistemas amigables*. Lima: Unión.

LePage, P. (2018). *Tu primera progressive web app*. Obtenido de https://developers.google.com/web/fundamentals/codelabs/your-first-pwapp/?hl=es#top_of_page

Lopez, M. (2015). *Definición de aplicación web*. Obtenido de <http://mialtoweb.es/defini>

cion-de-aplicacion-web/

Lopez, A. (2018). *Sistemas complejos. Ingeniería New*, 18.

Menzinsky, A., López, G., & Palacio, J. (2016). *Scrum manager*. Obtenido de http://www.scrummanager.net/files/sm_proyecto.pdf

MySQL. (2015). *About MySQL*. Obtenido de <https://www.mysql.com/about/>

MySQL. (2015). *MySQL Workbench*. Obtenido de <https://www.mysql.com/products/workbench/>

Pérez, K. (2012). *Taller de introducción a las bases de datos multimedia*. Obtenido de <https://kevinmichelleperez.files.wordpress.com/2012/02/taller-de-introduccc3b3n.pdf>

Perez Rivas, J. J. (2015). *Que es y como empezar con ionic framework*. Obtenido de <http://www.phonegapspain.com/que-es-y-como-empezar-con-ionic-framework/>

Pressman, R. (2018). *Ingeniería del software un enfoque práctico*. Obtenido de <http://cotana.informatica.edu.bo/downloads/ld-Ingenieria.de.software.enfoque.practico.7ed.Pressman.PDF>

Quispe Huiza, R. P. (2012). *Bolsa de trabajo carrera de informática*. Obtenido de http://repositorio.umsa.bo/bitstream/handle/123456789/9780/ilovepdf_merged.pdf?sequence=1

Raffo L., E., Ruiz L., E., Lorena L., P., Tupia D., E., Pérez Q., V., & Mayta H., R. (2002). *Implementación de la bolsa de trabajo on line en la facultad de Ingeniería Industrial - UNMSM. Industrial Data*, 44-51.

Salamon, A., Maller, P., Boggio, A., Mira, N., Perez, S., & Coenda, F. (2014). *La integración continua aplicada en el desarrollo de software en el ámbito científico-técnico*. Obtenido de <http://hdl.handle.net/10915/42358>

- Sauceda Gastélum, G. M. (2008). *Diseño de una bolsa de trabajo en el área de recursos humanos de una universidad*. Obtenido de http://biblioteca.itson.mx/dac_new/tesis/370_sauceda_gabriela.pdf
- Schwaber, K., & Sutherland, J. (2013). *La Guía de Scrum*. Obtenido de <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf>
- Stair, R., Reynolds, G. (2016). *Principios de Sistemas de Información*. Obtenido de https://issuu.com/cengagelatam/docs/stair_issuu
- Torres Huesca, M. A. (2011). *Propuesta de una aplicación (Web) móvil en la Facultad de Contaduría, Administración y Sistemas*. Obtenido de https://documentslide.org/the-philosophy-ofmoney.html?utm_source=universidad-veracruzana-tesina-licenciado-en-sistemas-computacionales-administrativos-maria-angelica-torres-huesca-maria-luisa-velasco-ramirez
- Universidad de Lima. (2014). *App Ulima está lista para ser descargada*. Obtenido de <http://www.ulima.edu.pe/ulima/noticias/app-ulima-esta-lista-para-ser-descargada>
- Yaya Gomez, B. M. (2015). *Sistema de Información para bolsa de trabajo en el Instituto de Educación Superior Tecnológico Público Argentina 2015*. Obtenido de <http://repositorio.unheval.edu.pe/handle/UNHEVAL/2164>

ANEXOS

ANEXO 1

Instrumento Aplicado: Encuesta los estudiantes de la UNTELS.

Para analizar la viabilidad del proyecto, se procedió a recabar información mediante una encuesta online. Según la Superintendencia Nacional de Educación Superior Universitaria en el periodo académico 2018 – I se matricularon un total de 2661 estudiantes en la UNTELS, 486 de Ingeniería Mecánica y Eléctrica, 571 de Ingeniería de Sistemas, 509 Administración de Empresas, 541 de Ingeniería Ambiental, 554 de Ingeniería Electrónica y Telecomunicaciones

Una vez conocida el tamaño de la población, se procedió a calcular la muestra, para ello utilizamos la formula estadística de “cálculo del tamaño de la muestra conociendo el tamaño de la población”, cuya formula es la siguiente.

$$n = \frac{N * Z^2 * p * q}{d^2 * (N - 1) + Z^2 * p * q}$$

En donde, n = tamaño de muestras, N = tamaño de la población, Z = nivel de confianza, p = probabilidad de éxito o proporción esperada, q = probabilidad de fracaso, d = precisión (Error máximo admisible en términos de proporción).

El valor de N es de 2661, tomamos un nivel de confianza del 95%, la probabilidad de éxito es 50%, la probabilidad de fracaso es 50%, el error máximo admisible es 5%. A continuación, se detalla el cálculo de la muestra.

$$n = \frac{2661 * 1.96^2 * 0.5 * 0.5}{0.05^2 * (2661 - 1) + 1.96^2 * 0.5 * 0.5}$$

$$n = 335.8068433$$

$$n \cong 336$$

Se determinó que la muestra es de 336 estudiantes.

La encuesta está compuesta de las siguientes preguntas:

¿A qué carrera perteneces?

- Administración de Empresas
- Ingeniería Ambiental
- Ingeniería Electrónica y Telecomunicaciones
- Ingeniería Mecánica y Eléctrica
- Ingeniería De Sistemas

¿Cómo calificaría el desempeño de la Bolsa de Trabajo de la UNTELS?

- Muy bueno
- Bueno
- Regular
- Malo

¿Considera que la Bolsa de Trabajo de la UNTELS tiene algún problema o carencia?

- Sí
- No

¿Con cuáles de los siguientes problemas se encontró?

- No puede hallar fácilmente la bolsa de trabajo en la página web de la UNTELS
- No sabe cuándo se realiza una nueva publicación o actualización
- No sabe cuáles son los requisitos para poder postular a una oferta laboral de la bolsa de trabajo
- La página web de la UNTELS no se encontraba disponible
- No dispongo de mucho tiempo para apersonarme y hacer tramites o entregar documentos
- No me notifican a tiempo si logré ser uno de los seleccionados

- Otros

¿Utiliza smartphone o algún otro dispositivo móvil?

- Sí
- No

¿Utilizaría una aplicación web que se encargue de gestionar la Bolsa de Trabajo

UNTELS?

- Sí
- No

¿Cuáles de estas funcionalidades le gustaría que tuviera dicha aplicación?

- Poder acceder de forma rápida y sencilla
- Notificarme cada vez que se realice una nueva publicación o actualización
- Ver el historial de publicaciones
- Poder descargar los documentos publicados
- Notificarme cuando la fecha de una convocatoria está cerca
- Notificarme oportunamente si logré ser uno de los seleccionados
- Otros

De esta encuesta se pudo recuperar las respuestas de 378 estudiantes de la UNTELS.

El primer objetivo de la encuesta era conocer que tan aceptado es la bolsa de trabajo de la UNTELS, para ello el criterio elegido fue el desempeño que este tiene para cada estudiante. Como se aprecia en la figura 28, el 52% de los encuestados consideran que la bolsa de trabajo tiene un desempeño regular, mientras que el 46% consideran que es malo y solo 2% considera que es buena. En base a ello se puede afirmar que la bolsa de trabajo cumple medianamente con el fin para el que fue creado

(brindar información, publicar convocatorias laborales, entre otros) aunque el nivel de satisfacción de los estudiantes encuestados no es el mejor.

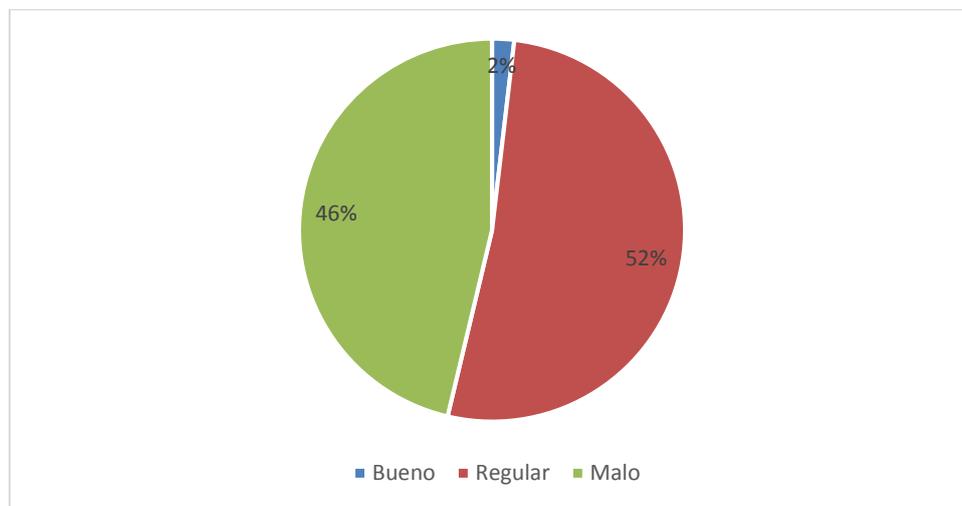


Figura 28. Valoración de la bolsa de trabajo según desempeño.
Fuente: Elaboración propia

El siguiente punto relevante de la encuesta fue determinar que problemas detectaron los estudiantes mientras usaban la bolsa de trabajo. La figura 29 muestra la cantidad de veces que un problema se presentó entre los estudiantes, siendo el problema “No sabe cuándo se realiza una nueva publicación o actualización” el más recurrente, este problema se debe en gran medida a que el estudiante no se entera cuando se realiza una nueva publicación en la bolsa de trabajo.

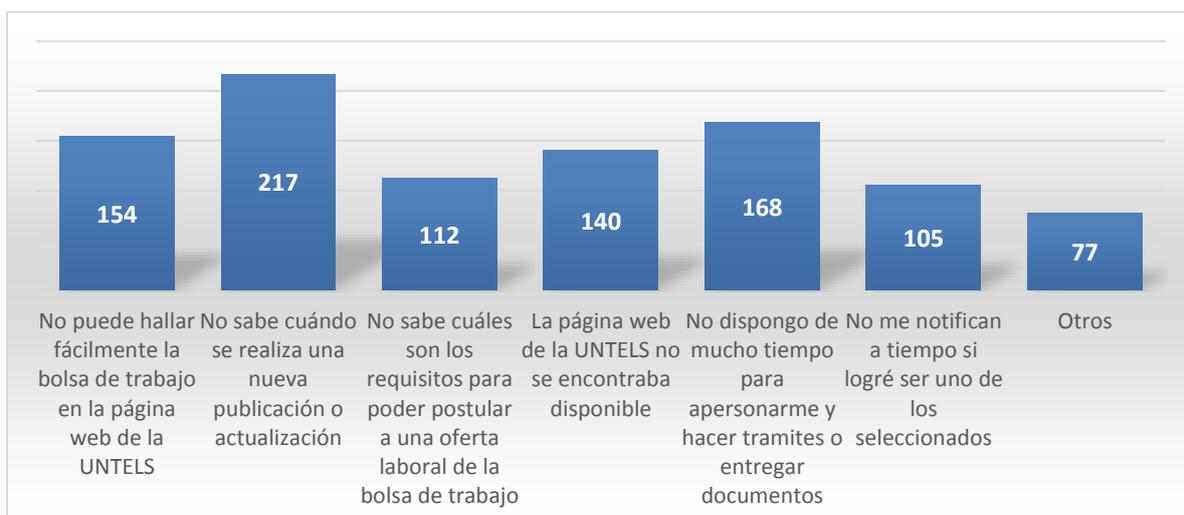


Figura 29. Problemas encontrados vs estudiantes.
Fuente: Elaboración propia

Las siguientes preguntas están más relacionadas al propósito de este trabajo, por ello era necesario conocer la penetración de los dispositivos móviles entre los estudiantes encuestados.

La primera pregunta busca segmentar al conjunto de estudiantes tomando como criterio si usan o no un smartphone u otro dispositivo móvil, tal como muestra la figura 30, el 93% los utilizan ya que estos son elementos indispensables en la vida cotidiana y un 7% no utilizan dispositivos móviles.

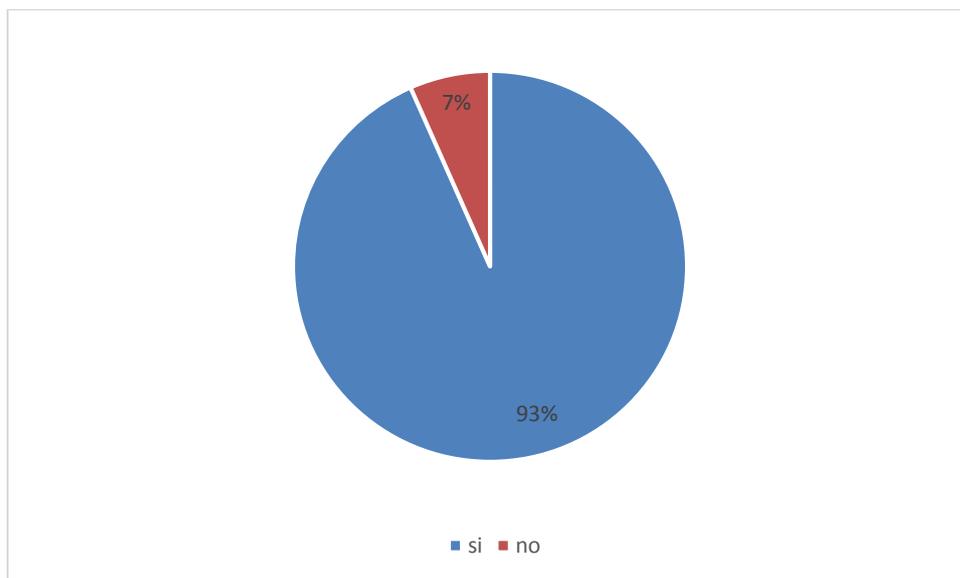


Figura 30. Porcentaje de estudiantes que usan Smartphone u otro dispositivo móvil.

Fuente: Elaboración propia

A partir de los 353 estudiantes que indicaron poseer un smartphone u otro dispositivo móvil se les consultó si estarían dispuestos a usar una aplicación que se encargue de gestionar la Bolsa de Trabajo y como muestra la figura 31, 95% la usarían. Aquí no se aplicó la segmentación por sistema operativo pues era más importante conocer la disposición de los estudiantes.

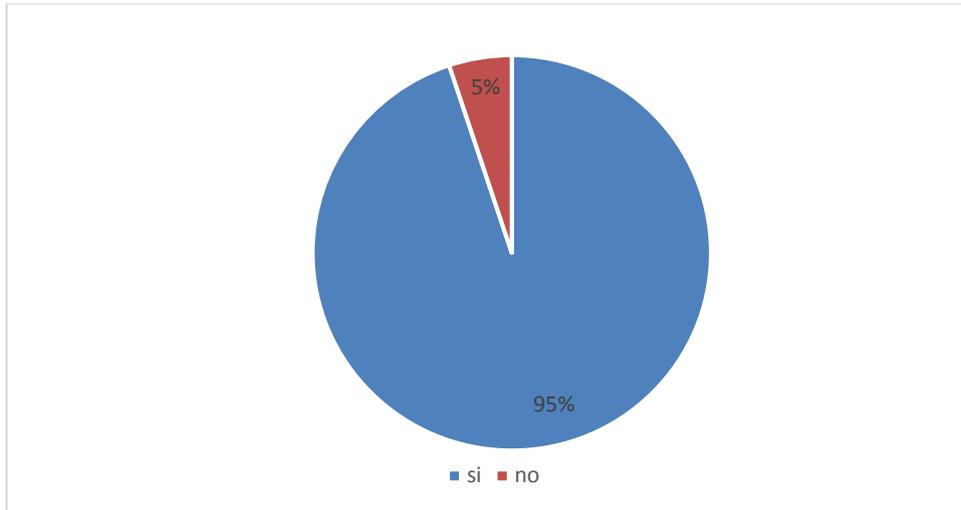


Figura 31. Estudiantes que utilizarían la aplicación.
Fuente: Elaboración propia

Finalizando el análisis de la encuesta, se preguntó a los estudiantes que eligieran entre múltiples funcionalidades que les gustaría ver en la aplicación prototipo. Como se presenta en la figura 32 las funcionalidades sugeridas fueron elegidas por la mayoría de encuestados, el cual es un buen indicador ya que la aplicación prototipo cumple con estas funcionalidades.

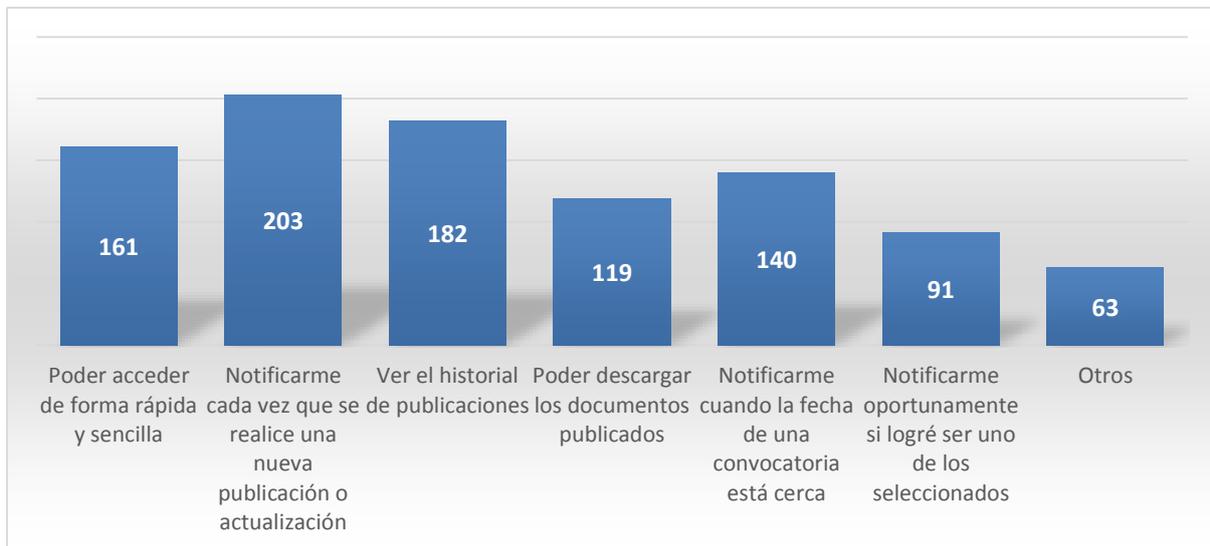


Figura 32. Funcionalidades vs estudiantes
Fuente: Elaboración propia

ANEXO 2

MANUAL DE USUARIO

Objetivo: El siguiente manual le permitirá aprender a usar todas las funcionalidades que presenta la aplicación web y móvil de la bolsa de trabajo en la UNTELS para estudiantes egresados.

Como acceder: Para ingresar a la página web o la aplicación móvil es necesario tener una conexión a internet, una vez ingresado a la aplicación se cargará el módulo de autenticación de la aplicación, dicho módulo será descrito en la siguiente sección.

Usuario: Estudiante / Egresado

Módulo de autenticación de usuarios

Una vez que se inicia la aplicación se mostrará la pantalla de autenticación del estudiante, en dicha pantalla debe de ingresar su usuario y contraseña, luego presionar el botón “Ingresar”. Una vez validada correctamente la autenticación del estudiante se mostrará la pantalla inicial de la aplicación.

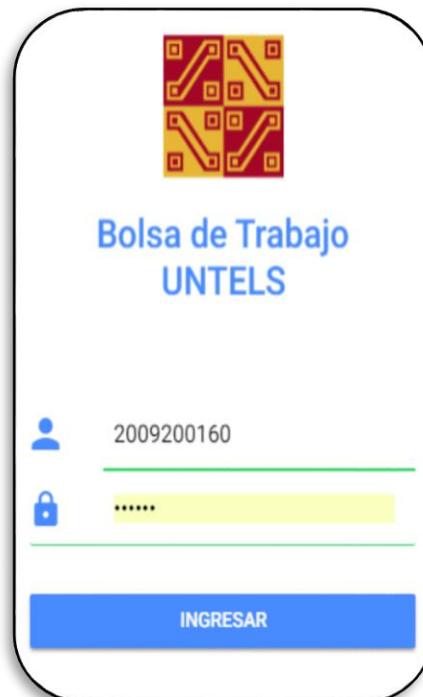


Figura 33. Login estudiante
Fuente: Elaboración propia

Menú de navegación

Una vez autenticado se accedera a la página de publicaciones en la parte inferior se muestra el menú de navegación con los botones de publicaciones, notificaciones y mi perfil.

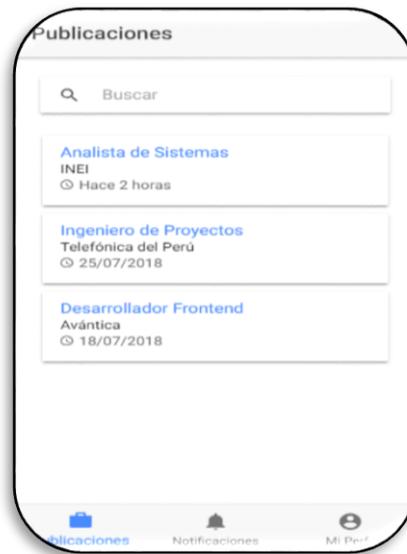


Figura 34. Menú principal
Fuente: Elaboración propia

Sección publicaciones

En esta sección se muestran las publicaciones de las ofertas laborales, los alumnos podrán visualizar las publicaciones y postular para la oferta laboral, seleccionando el botón de postular como se muestra a continuación.

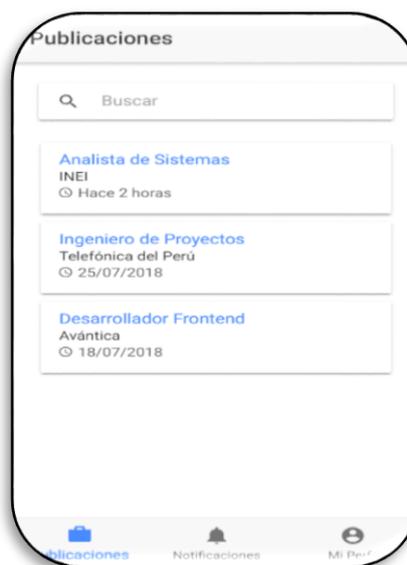


Figura 35. Publicaciones
Fuente: Elaboración propia



Figura 36. Detalle de publicación
Fuente: Elaboración propia

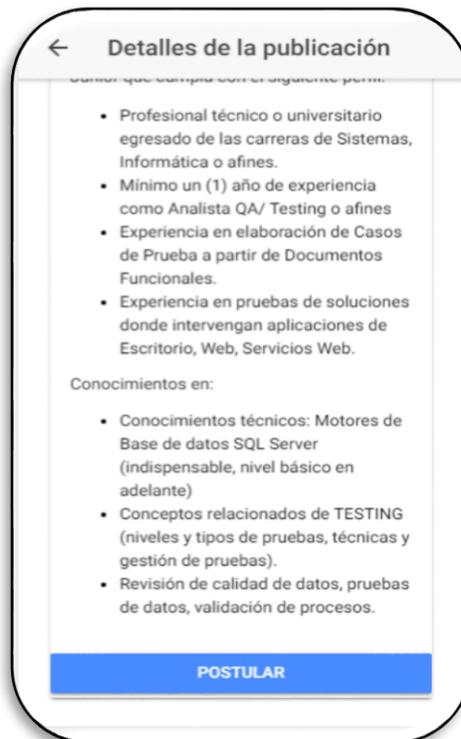


Figura 37. Postular
Fuente: Elaboración propia

Sección notificaciones

En la sección de notificaciones se muestran las actualizaciones y resultados de las publicaciones a las que postulaste.



Figura 38. Notificaciones
Fuente: Elaboración propia

Sección mi perfil

En esta sección se muestran la información de los estudiantes como el nombre, el código, la carrera, el ciclo, correo, teléfono, su curriculum vitae.

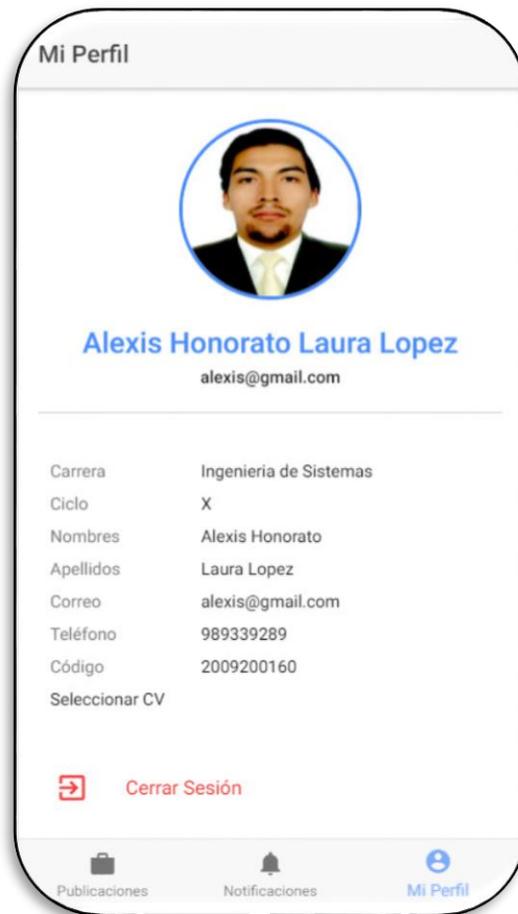


Figura 39. Mi perfil
Fuente: Elaboración propia

Usuario: Editor

Módulo de autenticación de editor

Una vez que se inicia la aplicación se mostrará la pantalla de autenticación de editor, en dicha pantalla debe de ingresar su usuario y contraseña, luego presionar el botón “Ingresar”. Una vez validada correctamente la autenticación del editor se mostrará la pantalla inicial de la aplicación.

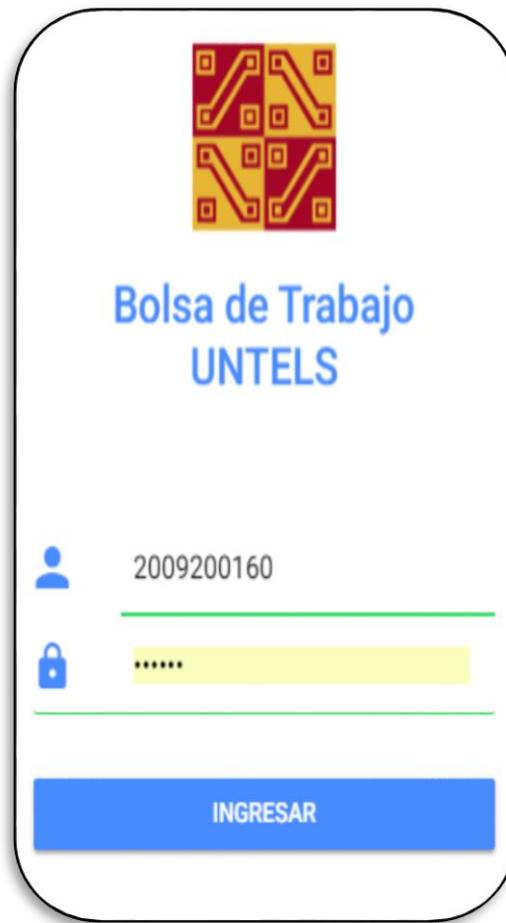


Figura 40. Login administrador
Fuente: Elaboración propia

Menú de Navegación

Una vez autenticado se accedera a la página de historial de publicaciones en la parte inferior se muestra el menú de navegación con los botones de historial de publicaciones y mi perfil.

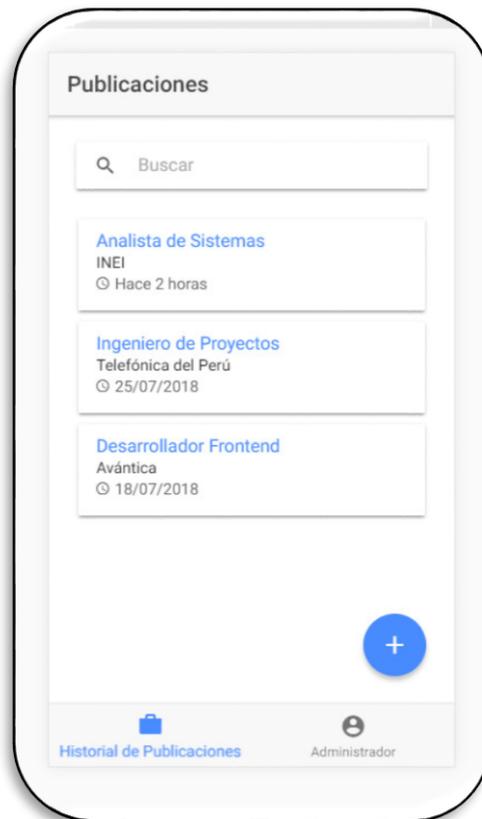


Figura 41. Menú principal
Fuente: Elaboración propia

Sección publicaciones

En esta sección se muestran las publicaciones que son realizados por la Oficina General de Tecnologías de la Información y Comunicación, también puede editar y eliminar, los alumnos podrán visualizar las publicaciones y postular para la oferta laboral, seleccionando el botón de postular como se muestra a continuación.

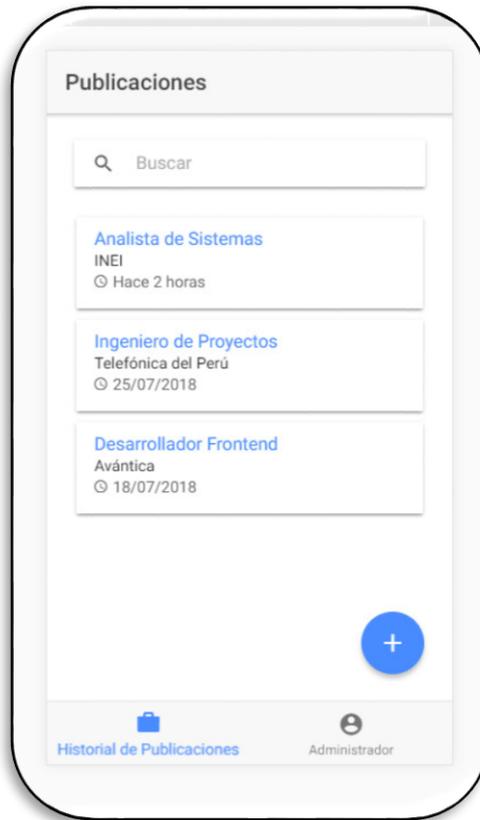


Figura 42. Historial publicaciones
Fuente: Elaboración propia



Figura 43. Nueva publicación
Fuente: Elaboración propia

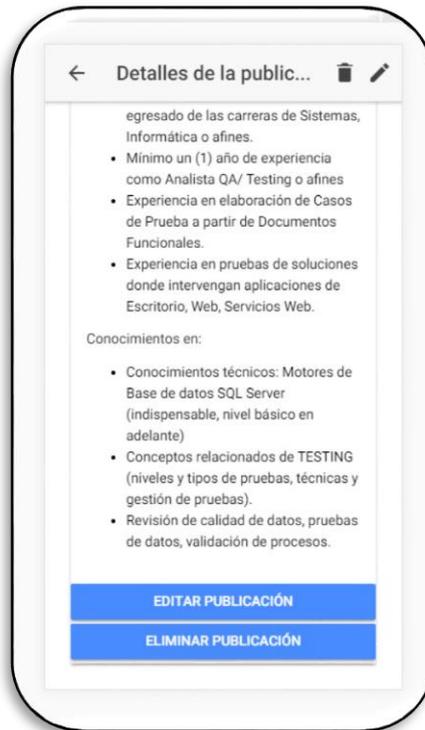


Figura 44. Editar y eliminar publicación
Fuente: Elaboración propia

Sección administrador

En esta sección se encuentra el botón de salir, para cerrar sesión en la aplicación.

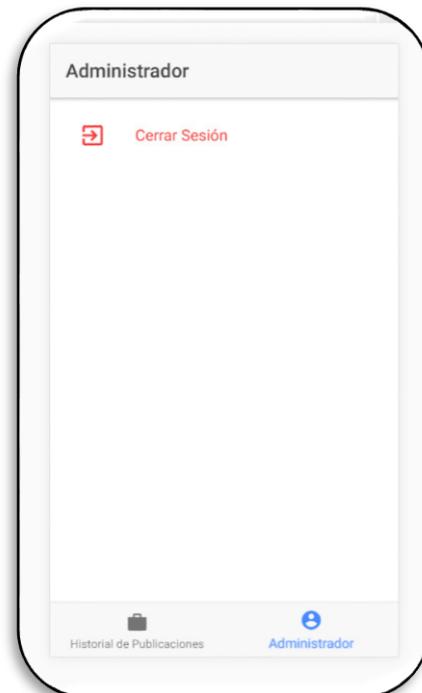


Figura 45. Administrador
Fuente: Elaboración propia