

SEP

SECRETARÍA DE
EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MÉXICO
Secretaría Académica, de Investigación e Innovación
Dirección de Posgrado, Investigación e Innovación

cenidet[®]
Centro Nacional de Investigación
y Desarrollo Tecnológico

Centro Nacional de Investigación y Desarrollo Tecnológico

Subdirección Académica

Departamento de Ciencias Computacionales

TESIS DE MAESTRÍA EN CIENCIAS

**LaborCheck: Sistema de Monitoreo Automático de Variables
Usuario-Computadora y de Interacción Social**

presentada por
Ing. Joan Misael Camaños Anacleto

como requisito para la obtención del grado de
Maestro en Ciencias de la Computación

Directora de tesis
Dra. Alicia Martínez Rebollar

Cuernavaca, Morelos, México. Diciembre de 2016.

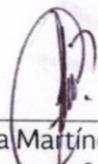
Cuernavaca, Morelos a 10 de noviembre del 2016
OFICIO No. DCC/228/2016

Asunto: Aceptación de documento de tesis

DR. GERARDO V. GUERRERO RAMÍREZ
SUBDIRECTOR ACADÉMICO
PRESENTE

Por este conducto, los integrantes de Comité Tutorial del **Ing. Joan Misael Camaños Anacleto**, con número de control M14CE055, de la Maestría en Ciencias de la Computación, le informamos que hemos revisado el trabajo de tesis profesional titulado "**LaborCheck: Sistema de monitoreo automático de variables usuario-computadora y de interacción social**" y hemos encontrado que se han realizado todas las correcciones y observaciones que se le indicaron, por lo que hemos acordado aceptar el documento de tesis y le solicitamos la autorización de impresión definitiva.

DIRECTORA DE TESIS



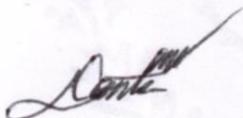
Dra. Alicia Martínez Rebollar
Doctora en Informática
7399055

REVISOR 1



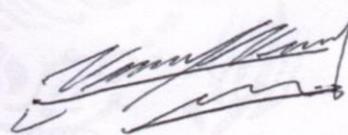
Dr. Noé Alejandro Castro Sánchez
Doctor en Ciencias de la
Computación
08701806

REVISOR 2



Dr. Dante Mújica Vargas
Doctor en Comunicaciones y
Electrónica
09131756

REVISOR 3



Dr. Luis Gerardo Vela Valdés
Doctor en Ciencias en Ingeniería
Electrónica
7980044

C.p. M.T.I. María Elena Gómez Torres - Jefa del Departamento de Servicios Escolares.
Estudiante
Expediente

NACS/lmz

Cuernavaca, Mor., 15 de noviembre de 2016
OFICIO No. SAC/322/2016

Asunto: Autorización de impresión de tesis

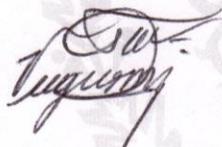
ING. JOAN MISAEL CAMAÑOS ANACLETO
CANDIDATO AL GRADO DE MAESTRO EN CIENCIAS
DE LA COMPUTACIÓN
PRESENTE

Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado "**LaborCheck: Sistema de monitoreo automático de variables usuario-computadora y de interacción social**", ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

ATENTAMENTE

"CONOCIMIENTO Y TECNOLOGÍA AL SERVICIO DE MÉXICO"



DR. GERARDO VICENTE GUERRERO RAMÍREZ
SUBDIRECTOR ACADÉMICO



SEP TecNM
CENTRO NACIONAL
DE INVESTIGACIÓN
Y DESARROLLO
TECNOLÓGICO
SUBDIRECCIÓN
ACADÉMICA

C.p. M.T.I. María Elena Gómez Torres.- Jefa del Departamento de Servicios Escolares.
Expediente

GVGR/mcr

Dedicatoria

Dedico esta tesis a mis padres Benigno Camaños Pérez y Margarita Anacleto Tenango porque a lo largo de mi vida profesional continúan apoyándome incondicionalmente tanto en el aspecto moral como en el económico. Sin embargo, sus consejos son los más importantes para elevar mis habilidades profesionales, los cuales me ayudaron a conseguir este grado de maestría.

Que Dios los bendiga y les de salud para que me sigan acompañando en los momentos más felices de mi vida porque gracias a ustedes mantengo este espíritu de superación para aportar tecnología que ayude a mejorar la calidad de vida de las personas.

Agradecimientos

Por su paciencia, su pasión por la investigación y principalmente por ofrecer temas de investigación que ayudan a detectar problemas que afectan a la sociedad en su desarrollo profesional y familiar a la Dra. Alicia Martínez Rebolgar.

Por el apoyo económico al Consejo Nacional de Ciencia y Tecnología (CONACYT) porque sin su apoyo no hubiese sido posible estudiar tiempo completo en un centro de investigación y adquirir esta formación de investigador.

Por darme la oportunidad de desarrollar mi proyecto de investigación dentro del departamento de ciencias computacionales en la especialidad de sistemas distribuidos al Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET).

Por las observaciones que permitieron mejorar la investigación tecnológica que realicé durante este periodo de estudios agradezco al comité revisor: Dr. Noé Alejandro Castro Sánchez, Dr. Dante Mújica Vargas y al Dr. Luis Gerardo Vela Valdés.

Por compartir conocimientos y experiencias profesionales a mis amigos y compañeros que conocí en el Centro Nacional de Investigación y Desarrollo Tecnológico: Virginia Ortiz Gómez, Guadalupe Candia Zavaleta, Wendy Aracely Sánchez Gómez, Marilú Cervantes, Estela Ortiz, Liliana Mérida Zagal, Jonathan Isai Moreno Cruz, Felipe Ojeda Cruz, Luis Obed Romero Nájera, Irving Ulises Herrera Molina, Richard Benítez y Alejandro de Jesús Guzmán Clemente, espero que posiblemente en un futuro colaboremos en artículos de investigación que realicemos con temas de interés común.

Por el ánimo que me brindo para continuar con mis estudios de maestría y ofrecerme un lugar cómodo para vivir durante los años en los que cursé la maestría a mi gran amiga Claudia Patricia Martínez Salgado.

Por ofrecer a los estudiantes temas de estudio actualizados útiles para el desarrollo profesional en todas las áreas de investigación y por tomar continuamente cursos de actualización al personal académico y administrativo del Centro Nacional de Investigación y Desarrollo Tecnológico.

Por estar al pendiente de mi situación académica y brindarme su apoyo a mis hermanos, tíos y amigos. Gracias por sus palabras de ánimo.

Resumen

El estrés laboral se presenta cuando un trabajador percibe que las exigencias laborales superan sus capacidades para mantenerlas bajo control. Las personas expuestas continuamente al estrés laboral, están propensas a padecer enfermedades graves tanto físicas como psicológicas.

Esta investigación presenta el sistema LaborCheck, un sistema de monitoreo capaz de recolectar y almacenar automáticamente la información de variables de interacción usuario-computadora y de interacción social.

Las variables de interacción usuario-computadora se recuperan por medio del uso del teclado, mouse e interacción entre las ventanas del sistema operativo Windows. Las variables de interacción social se recuperan por medio de la comunicación que generan los usuarios con sus contactos por medio de mensajes y llamadas del *Smartphone*.

El objetivo de esta investigación es recolectar y almacenar automáticamente información de variables usuario-computadora y de interacción social que pueda ser utilizada para la detección de estrés laboral.

Para cumplir con el objetivo principal, se desarrollaron dos aplicaciones que recolectan y almacenan remotamente la información. Estas aplicaciones se ejecutan automáticamente en los dispositivos de los usuarios que utilizan como medio de trabajo una computadora con sistema operativo Windows y como medio de comunicación un dispositivo móvil con sistema operativo Android.

Para evitar que la información no se pierda antes de enviarse al servidor remoto, se incluyó una base de datos local en cada una de las aplicaciones de monitoreo. La información se almacena temporalmente y en el momento que se detecta que existe una conexión a internet estos datos se envían al servidor de base de datos remoto.

Con el objetivo de que los usuarios que son monitoreados se sientan seguros al observar que las aplicaciones no recolectan ninguna información privada, se desarrolló una aplicación web que permite que los usuarios pueden consultar los valores de las variables que son recuperados por las aplicaciones de monitoreo. Esta aplicación permite filtrar por día los valores almacenados.

La evaluación de las aplicaciones desarrolladas se realizó por medio de un plan de pruebas en donde se comprueba la funcionalidad de los componentes incluidos en los módulos de cada aplicación. También se comparó la precisión del sistema LaborCheck con otro software que obtiene información de las mismas tecnologías incluidas en esta investigación demostrando que el funcionamiento del sistema LaborCheck es eficiente.

Abstract

Work stress occurs when workers perceive work demands exceed their capabilities to keep themselves under control. People who are exposed continuously to work stress are susceptible to suffer serious diseases both physical and psychological.

This research introduces the LaborCheck System, a system able to obtain and store automatically information of user-computer interaction variables and social interaction variables.

The user-computer interaction variables are obtained by using the keyboard, mouse, and windows interaction. The social interaction variables are obtained when users communicate with their contacts by messages and calls.

The main goal of this research is to obtain and store automatically information of user-computer variables and social interaction variables to be used for work stress detection in other research.

Two applications were developed to obtain and store remotely the information to reach the main goal. These applications run automatically in users' devices who use as work device a computer with Windows Operation System and as communication device a Smartphone with Android Operation System.

A local database was included in each one of the monitoring applications to avoid loss of information before sending it to the remote database which store temporally the information and after the system detects the internet connection, the data are sent to the remote database.

A web application was developed in order to demonstrate the users that none of their private information was obtained by the monitoring applications allowing the users check the variables values obtained. The web application allows the users filter the stored values per day.

The evaluation of the developed applications was carried out using a tests plan where the functionality of the included components in each application was tested. In addition, the LaborCheck accuracy was compared with other software that obtains the information of the same technologies included in this research demonstrating that LaborCheck system is efficient.

Índice general

Resumen	I
Abstract	III
Índice de figuras	IX
Índice de tablas	X
Capítulo 1. Introducción	3
1.1 Motivación.....	3
1.2 Planteamiento del problema	4
1.3 Solución propuesta	5
1.4 Objetivos	6
1.5 Estructura de la tesis	6
Capítulo 2. Marco Conceptual	11
2.1 Estrés	11
2.1.1 Estrés laboral	11
2.1.2 Factores desencadenantes de estrés.....	11
2.2 Interacción usuario-computadora.....	12
2.2.1 Monitoreo	12
2.3 .NET Framework.....	12
2.3.1 Hook	12
2.3.2 BackgroundWorker.....	13
2.4 Android.....	13
2.4.1 Servicios	13
2.4.2 Proveedores de contenidos	13
2.4.3 Receptores de mensajes de distribución.....	13
2.5 Servicio Web.....	14
2.5.1 SOAP	14
2.5.2 JSON	14
2.5.3 XML	14
2.6 Modelo cliente servidor	14
2.6.1 Cliente.....	15

2.6.2 Servidor	15
2.7 Patrón modelo vista controlador MVC	15
2.7.1 Modelo.....	15
2.7.2 Vista	15
2.7.3 Controlador.....	15
2.7.4 Framework MVC	16
2.8 Sistema gestor de bases de datos (SGBD).....	16
2.8.1 SQLite	16
2.8.2 MySQL	16
2.8.3 Transacciones SQL.....	16
Capítulo 3. Estado del Arte	19
3.1 Investigaciones que implementan tecnología relacionada con el mouse	20
3.1.1 MouseTracker: Software para el estudio del procesamiento mental incorporando un método computacional que utiliza el rastreo del mouse [10]	20
3.1.2 Capturando el interés del usuario a partir del registro de interacciones humano-computadora [12]	21
3.2 Investigaciones que implementan tecnología relacionada con el teclado	23
3.2.1 Identificación de estados emocionales usando la dinámica de tecleo [11].....	23
3.3 Investigaciones que implementan tecnología relacionada con mouse, teclado e interfaz gráfica de usuario	24
3.3.1 LoggerMan: una bitácora y herramienta de visualización para capturar el uso de la computadora [9].....	24
3.4 Investigaciones que implementan tecnología de comunicación por medio del <i>Smartphone</i>	26
3.4.1 Smartphone basado en predicción de estrés [13]	26
3.4.2 Modelando y descubriendo el comportamiento humano a través del registro de los datos detectados por medio del Smartphone [14].....	27
3.5 Conclusiones del estado del arte	29
Capítulo 4. Sistema LaborCheck	37
4.1 Metodología de solución	37
4.2 Esquema general del sistema LaborCheck.....	40
4.3 Aplicación de interacción usuario-computadora	41
4.3.1 Análisis de la aplicación de interacción usuario-computadora	41
4.3.2 Diseño de la aplicación de interacción usuario-computadora.....	43
4.3.3 Implementación de la aplicación de interacción usuario-computadora.....	46

4.4 Aplicación de interacción social.....	48
4.4.1 Análisis de la aplicación de interacción social.....	48
4.4.1 Diseño de la aplicación de interacción social.....	50
4.4.3 Implementación de la aplicación de interacción social.....	52
4.5 Aplicación web	55
4.5.1 Análisis de la aplicación web	55
4.5.2 Diseño de la aplicación web	57
4.5.3 Implementación de la aplicación web	57
4.6 Interfaz de comunicación	59
4.7 Base de datos remota.....	62
Capítulo 5. Pruebas y Resultados	65
5.1 Plan de pruebas para la aplicación de interacción usuario-computadora.....	65
5.1.1 Elementos de la prueba	65
5.1.2 Características probadas	65
5.1.3 Enfoque.....	66
5.1.4 Criterios de suspensión y requerimientos de reanudación	66
5.1.5 Requisitos ambientales	67
5.1.6 Responsabilidades.....	67
5.1.7 Nomenclatura	67
5.1.8 Casos de prueba	67
5.2 Plan de pruebas para la aplicación de interacción social	70
5.2.1 Elementos de la prueba	70
5.2.2 Características probadas	70
5.2.3 Enfoque.....	71
5.2.4 Criterios de suspensión y requerimientos de reanudación	71
5.2.5 Requisitos ambientales	71
5.2.6 Responsabilidades.....	72
5.2.7 Nomenclatura	72
5.2.8 Casos de prueba	72
5.3 Plan de pruebas para la aplicación web	75
5.3.1 Elementos de la prueba	75
5.3.2 Características probadas	75
5.3.3 Enfoque.....	76
5.3.4 Criterios de suspensión y requerimientos de reanudación	76
5.3.5 Requisitos ambientales	76

5.3.6 Responsabilidades.....	76
5.3.7 Nomenclatura	77
5.3.8 Casos de prueba	77
5.4 Pruebas con software relacionado.....	81
5.5 Resultados.....	83
Capítulo 6. Conclusiones y Trabajos Futuros	89
6.1 Conclusiones.....	89
6.2 Logros obtenidos.....	91
6.3 Trabajos futuros.....	91
Referencias	93
Anexo A. Descripción de la base de datos local de la aplicación de interacción usuario-computadora.....	99
Anexo B. Descripción de la base de datos local de la aplicación de interacción social.....	101
Anexo C. Descripción de la base de datos remota del sistema LaborCheck.....	105
Anexo D. Casos de prueba para la aplicación de interacción usuario-computadora.....	109
Anexo E. Casos de prueba para la aplicación de interacción social	115
Anexo F. Casos de prueba para la aplicación web.....	123

Índice de figuras

Figura 4.1 Metodología de solución	37
Figura 4.2 Esquema general del sistema LaborCheck.....	40
Figura 4.3 Diagrama de casos de uso de la aplicación de interacción usuario-computadora....	42
Figura 4.4 Arquitectura de la aplicación de interacción usuario-computadora	43
Figura 4.5 Diagrama entidad relación de la base de datos local para la aplicación de interacción usuario-computadora	45
Figura 4.6 Aplicación de interacción usuario-computadora.....	46
Figura 4.7 Diagrama de casos de uso de la aplicación de interacción social.....	49
Figura 4.8 Arquitectura de la aplicación de interacción social	50
Figura 4.9 Diagrama entidad relación de la base de datos local para la a aplicación de interacción social	52
Figura 4.10 Aplicación de interacción social	53
Figura 4.11 Diagrama de casos de uso de la aplicación web	56
Figura 4.12 Arquitectura de la aplicación web.....	57
Figura 4.13 Aplicación web.....	58
Figura 4.14 Diagrama entidad relación para la base de datos remota del sistema LaborCheck	62
Figura 5.1 Aplicación relacionada con la tecnología teclado	81
Figura 5.2 Aplicación relacionada con la tecnología mouse y ventanas	82
Figura 5.3 Aplicación relacionada con la tecnología de mensajes y llamadas	82
Figura 5.4 Actividad de las variables de la tecnología teclado	83
Figura 5.5 Actividad de las variables de la tecnología mouse.....	83
Figura 5.6 Actividad de las variables de la tecnología ventanas	84
Figura 5.7 Actividad de las variables de la tecnología mensajes.....	84
Figura 5.8 Actividad de las variables de la tecnología llamadas	85

Índice de tablas

Tabla 3.1 Tabla comparativa de los trabajos relacionados	30
Tabla 4.1 Requerimientos funcionales de la aplicación de interacción usuario-computadora	41
Tabla 4.2 Requerimientos no funcionales de la aplicación de interacción usuario-computadora	42
Tabla 4.3 Variables de interacción usuario-computadora.....	45
Tabla 4.4 Métodos programados para la aplicación de interacción usuario-computadora	47
Tabla 4.5 Requerimientos funcionales de la aplicación de interacción social.....	48
Tabla 4.6 Requerimientos no funcionales de la aplicación de interacción social.....	49
Tabla 4.7 Variables de interacción social.....	51
Tabla 4.8 Métodos programados para la aplicación de interacción social	54
Tabla 4.9 Requerimientos funcionales de la aplicación web	55
Tabla 4.10 Requerimientos no funcionales de la aplicación web	56
Tabla 4.11 Métodos programados para la aplicación web	59
Tabla 4.12 Servicios web para la gestión de usuarios	60
Tabla 4.13 Servicios web para la aplicación de interacción usuario-computadora.....	60
Tabla 4.14 Servicios web para la aplicación de interacción social.....	60
Tabla 4.15 Servicios web para la aplicación web	61

Capítulo 1

Introducción

Capítulo 1. Introducción

En este capítulo se describen algunos de los principales problemas que puede causar el estrés laboral tanto en la salud como en el aspecto económico de las compañías. Estos problemas dieron origen a la creación de este tema de investigación para ofrecer una solución que recupere información útil y que permita detectar el estrés laboral. Con el objetivo de plantear adecuadamente el tema de investigación se incluyen diferentes secciones, en las cuales se describe el planteamiento del problema, los objetivos, la solución propuesta y el contenido de la tesis que muestra la estructura general de esta investigación.

1.1 Motivación

Estudios realizados por la *American Psychological Association* demuestran que el nivel de estrés en los adultos aumentó en los últimos años. 75 % de los encuestados reportaron niveles de estrés de moderado a alto. De acuerdo con el *American Institute of Stress* 80% de los trabajadores se sienten estresados en sus puestos de trabajo y cerca de la mitad necesitan ayuda para manejar el estrés. *The Regus Group* reportó que los niveles de estrés laboral aumentaron en 6 de cada 10 trabajadores de las principales economías mundiales, siendo China uno de los países con mayores niveles de estrés [1].

De acuerdo con las encuestas realizadas por el Instituto Mexicano del Seguro Social (IMSS) tres cuartas partes de los empleados mexicanos padecen estrés laboral, mientras que, en China es 73% y en Estados Unidos 59%. México se reportó como el país con mayor porcentaje de estrés vinculado al trabajo, 75% de los empleados en el país padecen este problema y 25% de los infartos derivan de una enfermedad relacionada con dicho factor [2].

El estrés se define como un patrón de reacción compleja que a menudo tiene componentes psicológicos, cognitivos y conductuales [3]. El estrés laboral se manifiesta como un conjunto de reacciones nocivas, tanto físicas como emocionales, que ocurren cuando las exigencias del trabajo superan las capacidades, los recursos o las necesidades del trabajador, pudiendo originar enfermedad mental y hasta física [4].

Las reacciones que presentan las personas se manifiestan de diferentes maneras dependiendo de su resistencia psicológica, estrategia de afrontamiento y recursos ambientales. El estrés laboral ocurre cuando hay una incompatibilidad entre las exigencias del trabajo, el ambiente laboral, las capacidades, recursos y necesidades del trabajador. La exposición prolongada al estrés laboral ha sido vinculada a un patrón de consecuencias para la salud que incluye trastornos músculo-esqueléticos, agotamiento en el trabajo, depresión, lesiones, violencia en el lugar de trabajo y suicidio [5].

Las compañías están reconociendo cada vez más que los trabajadores mentalmente saludables son trabajadores más productivos. Los problemas de salud mental en el trabajo generalmente se manifiestan como problemas de desempeño: mayor ausentismo, menor productividad, mayor renovación de empleados, aumento en el número de días de discapacidad a corto y largo plazo. Por lo tanto, la falta de atención a la salud mental del trabajador y estrés laboral provocaría altos costos de operación [5].

Los sistemas de monitoreo tienen el propósito de recopilar información en tiempo real e indagar sobre situaciones o problemas que impidan el desarrollo de las personas y buscan detectar deficiencias, obstáculos o necesidades para intervenir oportunamente [6].

1.2 Planteamiento del problema

Un aspecto importante sobre el equilibrio laboral-personal es el número de horas que una persona trabaja. La evidencia sugiere que un horario de trabajo largo puede resultar perjudicial para la salud, poner en peligro la seguridad y aumentar el estrés. Las personas en México trabajan en promedio 2,226 horas al año, la mayor cifra de la Organización para la Cooperación y Desarrollo Económicos (OCDE), donde el promedio es de 1,765 horas [7].

La afectación en las empresas causada por estrés laboral impacta negativamente en las empresas de la siguiente forma: aumento del absentismo, menor dedicación al trabajo, aumento de la rotación del personal, deterioro del rendimiento y productividad, aumento de las prácticas laborales poco seguras y en las tasas de accidentes, aumento de las quejas de usuarios y clientes, efectos negativos en el reclutamiento de personal, aumento de los problemas legales ante las demandas presentadas y las acciones legales emprendidas por trabajadores que sufren estrés y deterioro de la imagen institucional por sus empleados [8].

El monitoreo automático en el lugar de trabajo es a la vez un medio necesario y eficaz para hacer frente a problemas de salud de los trabajadores y al crecimiento económico de las empresas. El enfoque principal de esta investigación está orientado principalmente en obtener automáticamente información de diferentes tecnologías que un usuario utiliza en su lugar de trabajo como son teclado, mouse, interacción con las ventanas del sistema operativo, mensajes y llamadas.

Existen varios trabajos [9-14], que desarrollaron sistemas de monitoreo automático para utilizarlos en diferentes aplicaciones. Estas aplicaciones utilizaron información de variables de las tecnologías que se incluyen en este tema de investigación. La recolección de variables de interacción usuario-computadora se obtuvo a través de computadoras de escritorio con sistema operativo Mac y Windows mientras que las variables de interacción social con dispositivos móviles con sistema operativo Android.

Con el uso del mouse se realizaron diferentes investigaciones obteniendo los eventos generados con este dispositivo. La detección de procesos mentales basados en la percepción gráfica y de sonidos. En [10], se desarrolló un software que almacena automáticamente las coordenadas en un archivo de texto para su posterior análisis. En [12], se recolectaron diferentes eventos del mouse como texto seleccionado y desplazamientos con la rueda del mouse para evaluar el interés del usuario sobre las páginas web que consulta. Esta información se almacenó en tiempo real en un servidor remoto.

Con el uso del teclado se analizaron diferentes emociones utilizando la dinámica del teclado. Algunas emociones que se analizaron fueron: estrés, ira, aburrimiento, confianza, distracción, concentración, ansiedad, tristeza, cansancio y nerviosismo. En [11], la información del teclado se recuperó utilizando librerías de Windows que interceptan los eventos del teclado.

Con la tecnología que recupera los cambios generados entre ventanas del sistema operativo se desarrolló un software que recupera estos cambios y detecta las ventanas y programas abiertos. En [9], se desarrolló un software para ofrecerlo a los investigadores como una herramienta de recolección que les permita analizar la información y detectar problemas de personalidad, cognición, etcétera.

Las tecnologías de comunicación relacionadas con mensajes y llamadas se emplearon para crear modelos que permitan identificar a un usuario y en predicción de estrés. En estas investigaciones [13, 14], se utilizaron diferentes componentes externos que recuperan información del registro de llamadas del sistema operativo Android y cada cierto tiempo almacenan la información en un archivo de texto y la envían a un servidor remoto.

Sin embargo, en algunas de las investigaciones anteriores se recolecta información que es privada para el usuario como: los caracteres de las teclas que se presionan, el contenido de los mensajes y el nombre de los contactos cuando se generan o reciben llamadas.

Otro aspecto que se presenta en los trabajos relacionados es que la información que se recolecta permanece en memoria durante largos periodos de tiempo antes de enviarse al servidor. En algunos casos no se envía a un servidor remoto, sino que se mantiene en el disco duro en archivos de texto. También se detectó que la mayoría del software que se implementó no ofrece una interfaz gráfica a los usuarios en donde puedan consultar la información recuperada por las aplicaciones de monitoreo.

1.3 Solución propuesta

En este trabajo de investigación se presenta un sistema de monitoreo automático que recupera la cantidad de interacciones clasificadas como interacciones usuario-computadora y de interacción social para que pueda utilizarse en la detección de estrés laboral. Los valores que se recuperan se almacenan localmente y posteriormente se envían a un servidor remoto.

Las variables de interacción usuario-computadora se recuperan dentro del sistema operativo Windows para detectar todas las interacciones que genera el usuario mientras se encuentra trabajando en su computadora, sin importar en que programa esté trabajando. Las tecnologías que se monitorean son teclado, mouse e interacciones entre ventanas.

Las variables de interacción social se recuperan dentro del sistema operativo Android de los empleados para monitorear la interacción que realizan con sus contactos personales o laborales. Las tecnologías a monitorear son mensajes y llamadas.

Los valores de las variables de interacción usuario-computadora y de interacción social se muestra gráficamente al usuario por medio de gráficas de series en donde se visualiza la información que se recolecta cada cierto tiempo. La información se visualiza correctamente en dispositivos de escritorio y en dispositivos móviles.

1.4 Objetivos

El objetivo general de esta investigación es recolectar y almacenar automáticamente información de variables usuario-computadora y de interacción social que pueda ser utilizada para la detección de estrés laboral. Para cumplir con el objetivo general se identificaron los siguientes objetivos específicos:

- Detectar automáticamente la información de las variables de interacción usuario-computadora por medio de las tecnologías: teclado, mouse e interacciones entre ventanas.
- Detectar automáticamente la información de las variables de interacción social por medio de las tecnologías: mensajes y llamadas.
- Almacenar automáticamente la información de variables usuario-computadora en una base de datos local que permanezca temporalmente en el disco duro de la computadora y que posteriormente esta información se envíe a una base de datos remota.
- Almacenar automáticamente la información de variables de interacción social en una base de datos local que permanezca temporalmente en la unidad de almacenamiento del dispositivo móvil y que posteriormente esta información se envíe a una base de datos remota.
- Mostrar gráficamente la información recolectada de las variables de interacción usuario-computadora y de interacción social en donde el usuario pueda visualizar los valores de todas las tecnologías: teclado, mouse, ventanas, mensajes y llamadas.

1.5 Estructura de la tesis

El contenido de la tesis presentada en este documento está organizado en diferentes capítulos que se muestran al lector con el fin de comprender la importancia de esta investigación, y que al mismo tiempo le permita generar nuevas ideas basadas en esta investigación.

- **Capítulo 2. Marco conceptual:** En este capítulo se mencionan los conceptos relacionados con el tema de investigación con referencias de instituciones y organizaciones especializadas en el tema. Estas referencias permiten que el lector comprenda los conceptos técnicos o desconocidos que se incluyen en el desarrollo de esta investigación.
- **Capítulo 3. Estado del Arte:** En este capítulo se presenta un resumen de los trabajos relacionados con respecto a sistemas de monitoreo automático para recuperar variables de tecnología relacionada con interacciones usuario-computadora y de interacción social. El resumen de los artículos se clasificó de acuerdo al tipo de tecnología que utilizan en cada investigación: teclado, mouse, ventanas, mensajes y llamadas.

- **Capítulo 4. Sistema LaborCheck:** En este capítulo se presentan los detalles técnicos del sistema de monitoreo LaborCheck. Se presentan las técnicas utilizadas para programar la aplicación de interacción usuario-computadora que se ejecuta en el sistema operativo Windows. También se describe el desarrollo de la aplicación de interacción social que se ejecuta en el sistema operativo Android y la aplicación web creada para mostrar al usuario la información recolectada por las aplicaciones de monitoreo.
- **Capítulo 5. Pruebas y Resultados:** En este capítulo se describen los planes de pruebas que se implementaron para probar la funcionalidad del sistema LaborCheck. Se mencionan tres planes de pruebas. El primero para la aplicación de interacción usuario computadora. El segundo para la aplicación de interacción social y el tercero para la aplicación web, cada uno de ellos se probaron en su entorno de implementación: Windows, Android y Navegador Web. También se incluyen pruebas para comparar la precisión de las aplicaciones de monitoreo con otras aplicaciones que monitorean los eventos relacionados con las mismas tecnologías. Finalmente se muestran diferentes gráficas con información de las variables que presentaron mayor actividad durante el periodo de pruebas.
- **Capítulo 6. Conclusiones y Trabajos Futuros:** En este capítulo se describen las contribuciones tecnológicas que aportó el sistema LaborCheck. Asimismo, se mencionan los logros obtenidos y las propuestas de las aplicaciones futuras que complementarían el sistema de monitoreo LaborCheck y así obtener los niveles de estrés laboral a partir del análisis de la información recolectada por medio de las aplicaciones de monitoreo automático.

Capítulo 2

Marco Conceptual

Capítulo 2. Marco Conceptual

En este capítulo se presentan los conceptos más relevantes que se desglosan alrededor del tema de investigación. En este contexto se incluyen conceptos relacionados con estrés, interacciones usuario-computadora, dispositivos móviles y conceptos técnicos relacionados con el desarrollo del sistema de monitoreo LaborCheck.

2.1 Estrés

El concepto de estrés proviene del término inglés "*stress*", que significa tensión provocada por situaciones agobiantes que generan reacciones orgánicas o trastornos psicológicos [4].

El estrés hace que el cerebro se ponga en guardia. La reacción del cerebro es preparar el cuerpo para la reacción defensiva. El sistema nervioso se despierta y las hormonas se despiertan para avivar los sentidos, acelerar el pulso, profundizar la respiración, y tensar los músculos. Esta respuesta (a veces llamada la respuesta de luchar o huir) es importante porque nos ayuda a defendernos contra situaciones amenazantes [15].

Los episodios de estrés que duran poco o están infrecuentes representan poco riesgo. Pero cuando las situaciones estresantes continúan no resueltas, se queda el cuerpo en un estado constante de activación, lo que aumenta la tasa de desgaste a los sistemas biológicos [15].

2.1.1 Estrés laboral

El estrés relacionado con el trabajo es un patrón de reacciones psicológicas, emocionales, cognitivas y conductuales ante ciertos aspectos extremadamente abrumantes o exigentes en el contenido, organización y ambiente de trabajo. Cuando las personas experimentan estrés laboral, a menudo se sienten tensas y angustiadas y sienten que no pueden hacer frente a las situaciones [16].

2.1.2 Factores desencadenantes de estrés

Entre los factores desencadenantes del estrés (conocidos como estresantes) pueden mencionarse los estímulos internos y externos que, de manera directa o indirecta, provocan la desestabilización en el equilibrio dinámico del organismo [4]:

- Situaciones que exigen una rápida solución.
- Exceso de trabajo.
- Responsabilidades familiares.
- Estímulos ambientales dañinos (ruido, contaminación, etcétera).
- Percepciones de amenaza.
- Alteración de las funciones fisiológicas (enfermedades, adicciones, etcétera).
- Aislamiento y encierro.
- Presión grupal o laboral.
- Frustración.

2.2 Interacción usuario-computadora

La interacción entre humanos y computadoras, es un área de investigación multidisciplinaria, esta disciplina investiga y trata todos los aspectos relacionados con las interfaces entre los humanos y las computadoras [17].

Los seres humanos interactúan con computadoras a través de una interfaz de usuario. Esta interacción incluye software como lo que visualizamos en pantalla y hardware como mouse, teclado y otros dispositivos periféricos. El estudio de la interacción humano-computadora se enfoca en la satisfacción del usuario, la atención de la interacción humano-computadora es importante, porque una interfaz pobre puede ser difícil para los usuarios sin beneficiarse incluso de los sistemas más sencillos [18].

2.2.1 Monitoreo

El monitoreo es un seguimiento continuo de lo que está sucediendo durante la implementación de algún sistema, en el que se analizan cómo evolucionan ciertos indicadores. El monitoreo de estos indicadores recopila y reporta sistemáticamente las tendencias con el propósito de informar a los responsables sobre las situaciones que se generan con el programa. La evaluación intermedia, indaga sobre las condiciones que pueden facilitar o impedir el logro de los objetivos. Examina en forma periódica que está sucediendo para responder a preguntas específicas relacionadas con la ejecución y los resultados e intervenir oportunamente [6].

2.3 .NET Framework

Es un entorno de ejecución que proporciona diversos servicios a las aplicaciones en ejecución. Consta de dos componentes principales: *Common Language Runtime (CLR)*, que es el motor de ejecución que controla las aplicaciones y la biblioteca de clases de *.NET Framework*, que proporciona una biblioteca de código probado y reutilizable al que pueden llamar los desarrolladores desde sus propias aplicaciones [19].

2.3.1 Hook

Es un mecanismo por el que una aplicación puede interceptar eventos como mensajes, acciones con el mouse y pulsaciones del teclado. Una función que intercepta un tipo particular de evento es conocida como *hook procedure*. Un *hook procedure* puede actuar en cada evento que se recibe, modificar o descartar el evento [20].

El sistema soporta diferentes tipos de *hooks*, cada tipo proporciona acceso a un aspecto diferente de su mecanismo de manejo de mensajes. Por ejemplo, una aplicación puede usar el *WH_MOUSE* hook para monitorear el tráfico de los mensajes del mouse [20].

2.3.2 BackgroundWorker

La clase *BackgroundWorker* permite ejecutar una operación en un hilo separado. Operaciones que consumen mucho tiempo como descargas, transacciones de bases de datos que pueden provocar que la interfaz del usuario se detenga y no responda. Cuando se desea una interfaz de usuario con respuestas y debe llevar a cabo las operaciones que consumen mucho tiempo, esta clase proporciona una solución conveniente [21].

2.4 Android

Es un sistema operativo y una plataforma de software, basado en Linux para teléfonos móviles. Android permite trabajar en un entorno de trabajo de Java, programando aplicaciones sobre una máquina virtual *Dalvik* (una variación de la máquina virtual de Java con compilación en tiempo de ejecución). Dado que Android está basado en el núcleo de Linux, tiene acceso a sus recursos, pudiendo gestionarlos, gracias a que se encuentra en una capa por encima del Kernel, accediendo así a recursos como los controladores de pantalla, cámara, memoria flash, etc [22].

2.4.1 Servicios

Los servicios son tareas no visibles que se ejecutan siempre en segundo plano, incluso cuando la actividad asociada no se encuentra en primer plano. Tienen un hilo propio, lo que permite llevar a cabo cualquier tarea y no necesita interfaz [22].

2.4.2 Proveedores de contenidos

Estos proveedores en inglés llamados *content provider*, se encargan de que la aplicación pueda acceder a la información que necesita, siempre que se haya declarado el correspondiente *provider* en el *AndroidManifest*. Implementan una Interfaz, pero se comunican con ella a través de la clase *ContentResolver*. Cada vez que se usa un *ContentResolver*, se activa un *ContentProvider*. Para obtener los datos necesarios, es necesario conocer la URI (Identificador) del dato, los datos que contiene, y los tipos de campos. Con esto es suficiente para poder llamar al método *ContentResolver.query()* y recuperar los datos requeridos [22].

2.4.3 Receptores de mensajes de distribución

También llamados *broadcast receiver*, son los encargados de reaccionar ante los eventos ocurridos en el dispositivo, ya sean generados por el sistema o por una aplicación externa. No tienen interfaz, pero pueden lanzar una actividad por medio de un evento. La clase que defina estos componentes heredará de la clase *BradCastReceiver* [22].

2.5 Servicio Web

Un servicio web es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma [23].

2.5.1 SOAP

Es un protocolo para intercambio de información en un ambiente descentralizado y distribuido. Es un protocolo basado en XML compuesto por tres partes: una envoltura que define un marco de trabajo para describir lo que está dentro del mensaje y como procesarlo, un conjunto de reglas para expresar instancias de los tipos de datos definidos en la aplicación y una notación para representar llamadas y respuestas a procedimientos remotos [24].

2.5.2 JSON

JSON es un formato de intercambio de datos ligero. Es fácil de leer y escribir. Es un formato de texto que es completamente independiente del lenguaje de programación, pero usa las convenciones que son familiares para los programadores que usan los programas de la familia de C incluyendo C++, C#, Java, JavaScript, Perl, Python y muchos otros. Estas propiedades permiten que JSON sea un lenguaje ideal para el intercambio de datos [25].

2.5.3 XML

XML es un lenguaje etiquetado extensible que actúa principalmente en el intercambio de una gran variedad de datos. Es un lenguaje muy similar a HTML, pero su función principal es describir datos y no mostrarlos como es el caso de HTML. XML es un formato que permite la lectura de datos a través de diferentes aplicaciones. Las tecnologías XML son un conjunto de módulos que ofrecen servicios útiles a las demandas más frecuentes por parte de los usuarios. XML sirve para estructurar, almacenar e intercambiar información [26].

2.6 Modelo cliente servidor

Es una arquitectura distribuida que permite a los usuarios finales obtener acceso a la información en forma transparente aún en entornos multiplataforma. En el modelo cliente-servidor, el cliente envía un mensaje solicitando un determinado servicio a un servidor (crea la petición), y este envía uno o varios mensajes con la respuesta (provee el servicio). En un sistema distribuido cada máquina puede cumplir el rol de servidor para algunas tareas y el rol de cliente para otras [27].

2.6.1 Cliente

Es el proceso que permite al usuario formular los requerimientos y pasarlos al servidor. El cliente maneja todas las funciones relacionadas con la manipulación y despliegue de datos, por lo que están desarrollados sobre plataformas que permiten construir interfaces gráficas de usuario (GUI), además de acceder a los servicios distribuidos en cualquier parte de una red [27].

2.6.2 Servidor

Es el proceso encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él. El servidor maneja todas las funciones relacionadas con la mayoría de las reglas del negocio y los recursos de datos [27].

2.7 Patrón modelo vista controlador MVC

Es un paradigma que divide las partes que conforman una aplicación en el modelo, las vistas y los controladores, permitiendo la separación por separado de cada elemento, garantizando así la actualización y mantenimiento del software de forma sencilla y en un reducido espacio de tiempo [28].

2.7.1 Modelo

El modelo es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El modelo no tiene conocimiento específico de los controladores o de las vistas, ni siquiera contiene referencias a ellos. Es el propio sistema el que tiene encomendada la responsabilidad de mantener enlaces entre el modelo y sus vistas, y notificar a las vistas cuando cambia el modelo [28].

2.7.2 Vista

La vista es el objeto que maneja la representación visual de los datos representados por el modelo. Genera una representación visual del modelo y muestra los datos al usuario. Interactúa preferentemente con el controlador, pero es posible que trate directamente con el modelo a través de una referencia al propio modelo [28].

2.7.3 Controlador

El controlador es objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el modelo, centra toda la interacción entre la vista y el modelo. Cuando se realiza algún cambio, entra en acción, bien sea por cambios en la información del modelo o por alteraciones de la vista. Interactúa con el modelo a través de una referencia al propio modelo [28].

2.7.4 Framework MVC

Los patrones MVC cumplen perfectamente el fin particular de cualquier *Framework*: proveer una estructura bien definida que de soporte a un proyecto que ayude a que el proyecto sea organizado y bien desarrollado [28].

El uso de los *Frameworks* basados en este patrón permite tener una separación lógica y física de los componentes de la aplicación, por un lado, se tienen los modelos, por otro las vistas y por otro los controladores. De esta forma los desarrolladores de la aplicación pueden centrarse en la parte que les toca, ya sea como diseñadores en las vistas, o como programadores de los modelos del negocio [28].

2.8 Sistema gestor de bases de datos (SGBD)

Un sistema gestor de bases de datos consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos. La colección de datos, normalmente denominada base de datos, contiene información relevante para una organización [29].

El objetivo principal de un SGBD es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea tanto práctica como eficiente. Los sistemas de bases de datos se diseñan para gestionar grandes cantidades de información [29].

2.8.1 SQLite

SQLite es una biblioteca que implementa un motor de base de datos auto-contenido, sin servidor, sin necesidad de configuración y transaccional. SQLite no tiene un proceso independiente de servidor. SQLite lee y escribe directamente en archivos de disco ordinarios. Una base de datos completa con múltiples tablas, índices y vistas está contenida en un sólo archivo de disco [30].

2.8.2 MySQL

MySQL es un sistema relacional de base de datos. MySQL es un sistema cliente-servidor en donde hay un servidor de base de datos MySQL y muchos clientes (aplicaciones), los cuales se comunican con el servidor para consultar datos, guardar cambios, etcétera. MySQL soporta como lenguaje de base de datos SQL. SQL es un lenguaje estandarizado para la consulta y actualización de datos y para la administración de una base de datos [31].

2.8.3 Transacciones SQL

Una transacción es una unidad lógica de trabajo que ejecuta un conjunto de sentencias como una sola sentencia SQL. Una transacción debe acabar con alguna de las siguientes sentencias: *COMMIT* o *ROLLBACK*, la diferencia entre *COMMIT* y *ROLLBACK* es que la sentencia *COMMIT* confirma todos los cambios producidos y la sentencia *ROLLBACK* deshace todos los cambios producidos en la base de datos y la deja como estaba antes del inicio de la transacción [32].

Capítulo 3

Estado del Arte

Capítulo 3. Estado del Arte

En este capítulo se muestra el resumen de los artículos relacionados con el tema de investigación que se abordó. Algunos investigadores se enfocan en problemas psicológicos que pueden ser detectados por medio de las interacciones que generan los usuarios, utilizando dispositivos tecnológicos como teclado, mouse y *Smartphone*.

En cambio, otros investigadores analizan la experiencia percibida por los usuarios al navegar sobre aplicaciones web. El interés de las páginas web es evaluado por medio de las interacciones generadas con el teclado y mouse, es decir el desplazamiento entre las páginas web, cantidad de clics generados etc.

Los artículos de investigación se resumen de acuerdo a diferentes criterios de análisis:

Descripción general. Contiene los aspectos generales acerca de los problemas psicológicos que se abordan y la tecnología incluida en cada artículo.

Arquitectura. Contiene la descripción de la estructura de los programas o módulos desarrollados, la comunicación entre los procesos programados y el software que se utilizó para la codificación.

Evaluación. Contiene la descripción de las pruebas que se aplicaron para comprobar la integridad de los sistemas de monitoreo, también se mencionan los experimentos que se emplearon para evaluar los procesos psicológicos.

Conclusiones. Contiene las principales contribuciones que aportó el trabajo de investigación como sistemas de software o modelos de clasificación.

Los artículos que incorporan investigación relacionada con interacciones usuario-computadora se clasifican en tres categorías: investigaciones que implementan tecnología relacionada con el mouse, investigaciones que implementan tecnología relacionada con el teclado e investigaciones que implementan tecnología relacionada con mouse, teclado e interfaz gráfica de usuario.

Los artículos que incluyen investigación relacionada con la interacción social por medio de mensajes y llamadas se clasifican en una sola categoría: investigaciones que implementan tecnología de comunicación por medio del *Smartphone*.

3.1 Investigaciones que implementan tecnología relacionada con el mouse

3.1.1 *MouseTracker: Software para el estudio del procesamiento mental incorporando un método computacional que utiliza el rastreo del mouse [10]*

Descripción general

En esta investigación se desarrolla un software que permite a los investigadores reconocer en tiempo real algunos procesos mentales por medio del rastreo del mouse. El software permite analizar la percepción de estereotipos asociados y la percepción de sonidos. Almacena las trayectorias a través de la grabación de las coordenadas x , y del puntero del mouse. Los investigadores utilizan estas trayectorias para evaluar la orientación de las respuestas seleccionadas por los usuarios por medio de la curvatura que presenta la trayectoria que el usuario sigue desde el punto donde ubica el puntero del mouse para iniciar la prueba hasta seleccionar alguna de las opciones de respuesta disponibles sobre la interfaz gráfica.

Arquitectura

Este software se estructura en tres módulos:

1. Configuración del software. El programa permite agregar o eliminar las respuestas psicológicas que se muestran en la interfaz gráfica. Asimismo, permite seleccionar un archivo de texto plano separado por comas para almacenar la información obtenida por cada usuario. El archivo incluye diferentes campos: las coordenadas x , y del puntero del mouse, la hora de cada movimiento del puntero, la hora de inicio de la prueba, y la respuesta seleccionada de cada usuario.
2. Ajuste de parámetros gráficos. En este módulo se incorpora el ajuste automático del espacio entre las coordenadas de cada trayectoria, se utiliza interpolación lineal para incluir cien pares de coordenadas en cada trayectoria. Asimismo, el programa permite personalizar el tamaño de los botones, cambiar la ubicación de los elementos gráficos dentro de la interfaz, cambiar el tipo y tamaño de fuente e incluir imágenes y sonidos.
3. Módulo analítico. En este módulo se analizan las trayectorias generadas por los usuarios, utilizando medidas como el área bajo la curva y desviación máxima. También se calcula la velocidad y la aceleración promedio de las trayectorias de todos los usuarios.

El programa detecta en segundo plano los movimientos del puntero del mouse y las acciones realizadas sobre la interfaz gráfica, enviando los datos recolectados al archivo de texto diseñado en el módulo de configuración. La información de los movimientos del mouse es obtenida programando llamadas directas hacia la interfaz de programación de *Windows*.

Evaluación

En esta investigación se incluyen dos tipos de evaluación, la primera consiste en evaluar la precisión de los datos del software y la segunda en llevar a cabo los experimentos con el software desarrollado.

Para medir la precisión del software MouseTracker, doce participantes completaron tres tareas, la primera vez ejecutándolas bajo el software MouseTracker y la segunda ejecutándolas bajo el software DirectRT, la información recolectada fue muy similar en aparecía y tiempo.

Para medir los estereotipos personales, se realizaron 24 pruebas, las pruebas mostraron gráficamente rostros femeninos y masculinos típicos y atípicos, con el objetivo de categorizar los rostros, debido a que los rostros con apariencia bondadosa se categorizaron como rostros femeninos y rostros con apariencia agresiva para rostros masculinos. Sin embargo, las trayectorias del mouse fueron continuamente atraídas hacia la categoría del sexo opuesto antes de establecerse en la categoría correcta.

Por otro lado, para medir la percepción correcta de los sonidos, la prueba consistió en presentar a los participantes la imagen de una vela en la esquina superior izquierda y la imagen de un dulce en la esquina superior derecha, los participantes mostraron una curvatura sobresaliente hacia el distractor *candy* antes de establecerse en la alternativa correcta *candle*, mostrando un ligero traslape al detectar estos fonemas del idioma inglés.

Conclusiones

La contribución de esta investigación es un software que permite almacenar y analizar las trayectorias generadas por los usuarios durante pruebas psicológicas mostradas en una interfaz gráfica. Con este software los investigadores pueden evaluar la orientación que presenta un grupo de usuarios hacia ciertos estereotipos y percepciones con un método sensible al tiempo, práctico y económico.

3.1.2 Capturando el interés del usuario a partir del registro de interacciones humano-computadora [12]

Descripción general

En esta investigación se implementa un modelo para inferir sobre la utilidad de los documentos web interpretando las acciones generadas por el mouse. Este modelo es implementado en un sistema de posicionamiento web llamado YAR, el cual reposiciona las páginas web que se obtienen por medio de un motor de búsqueda personalizado. El modelo calcula los valores de relevancia por medio de las interacciones generadas con el mouse para reposicionar las páginas web. Las interacciones generadas por el usuario se recolectan automáticamente a través de una extensión que pueden ser instalada en cualquier navegador web.

Arquitectura

El modelo implementado es embebido en el sistema YAR, este sistema utiliza la arquitectura cliente-servidor que se compone de dos módulos.

1. Módulo de registro. Este módulo detecta el comportamiento del usuario mientras navega por la web, enviando la información de los eventos capturados al servidor. Rastrea el tiempo de permanencia en las páginas web, los movimientos del cursor, eventos con la rueda del mouse y el texto seleccionado. Este módulo es desarrollado utilizando la tecnología AJAX.
2. Módulo analizador. Este módulo calcula las métricas definidas en el modelo: tiempo de permanencia, se obtiene a través de la diferencia de tiempo de visita y abandono de una página web que puede depender del número total de palabras que contenga la página web, del lenguaje y de la edad del usuario; tasa de lectura, se considera el número de letras que se recorren con el cursor del mouse, total de palabras del documento y número total de palabras seleccionadas; y tasa de desplazamientos, que incluye el número de desplazamientos por minuto. El análisis de regresión se utiliza para calcular los pesos óptimos de las métricas.

Evaluación

Las extensiones desarrolladas que recuperan las interacciones del mouse fueron instaladas en el navegador web de 26 participantes con edades de 22 a 31 años. Los participantes dieron lectura a documentos web personalizados que fueron previamente incluidos en el sistema YAR.

Los resultados obtenidos por el modelo se compararon con cuestionarios, donde los usuarios calificaron la utilidad de la página web por medio de la escala LIKER con valores de 1 a 5. Estos cuestionarios fueron incluidos en el sistema YAR y mostrados al usuario por medio de una página web.

Los documentos web que fueron consultados por medio del motor de búsqueda incluido en el sistema YAR, se reordenaron de acuerdo a los valores de relevancia calculados por medio de las interacciones del mouse. Los documentos web que se recuperaron mostraron de 1 a 5 estrellas al lado de cada enlace de la página que indicó la relevancia de interés del usuario sobre el documento web. El modelo fue capaz de predecir el interés del usuario con un nivel aceptable de precisión.

Conclusiones

Esta investigación contribuye con un modelo que permite inferir sobre el interés que un usuario tiene acerca de los documentos web, tomando en cuenta las acciones que el usuario genera cuando consulta cada una de las páginas que se recuperan por medio del motor de búsqueda incluido en el sistema YAR. Las herramientas de monitoreo incluidas en este sistema son programadas por medio de extensiones ligeras que se instalan en un navegador web, estas extensiones permiten recuperar los valores de las variables del mouse sin interrumpir la navegación del usuario.

3.2 Investigaciones que implementan tecnología relacionada con el teclado

3.2.1 Identificación de estados emocionales usando la dinámica de tecleo [11]

Descripción general

En esta investigación se desarrolla un software para detectar los estados emocionales que se presentan a través del ritmo de la escritura, utilizando el teclado de una computadora común. Esta investigación considera los lapsos de tiempo transcurridos entre las teclas presionadas y las teclas liberadas. Los modelos de clasificación se generan utilizando árboles de decisión para modelar 15 estados emocionales: ira, aburrimiento, confianza, distracción, emoción, concentración, frustración, felicidad, indecisión, nerviosismo, ansiedad, relajación, tristeza, estrés y cansancio.

Arquitectura

El software implementado en esta investigación se estructura en tres módulos:

1. Módulo de recolección de datos. Las pulsaciones del teclado se recuperan automáticamente en segundo plano, la librería de bajo nivel Hook de Windows se utiliza para recuperar los eventos del teclado keyUp y keyDown incluyendo la fecha y hora en que ocurrieron estos eventos. El uso de esta librería permite recuperar las pulsaciones generadas sobre el teclado mientras se está trabajando en cualquier aplicación de Windows. Los datos recuperados permanecen en memoria durante el tiempo en el que duran las pruebas y después se envían a un servidor remoto.
2. Interfaz gráfica. Dentro de la interfaz gráfica se incluye una opción en donde el usuario puede detener la recolección de su información en cualquier momento, por ejemplo: cuando utiliza el teclado para introducir contraseñas. Asimismo, se desarrolla un formulario para introducir textos personalizados, deshabilitando las opciones de copiar y pegar. Finalmente, se incluye un formulario que despliega un cuestionario con 15 estados emocionales.
3. Módulo de procesamiento de datos. Las variables que se consideran para crear los modelos de clasificación se organizan en tres categorías: tiempo de pulsación de teclas individuales (duración entre la tecla liberada y la tecla presionada), tiempo de pulsaciones agrupadas en dígrafos (duración entre la primera y segunda tecla presionada; y duración entre la primera tecla liberada y la segunda tecla presionada) y tiempo de pulsaciones agrupadas en trígrafos (duración entre la primera y segunda tecla presionada; duración entre la primera tecla liberada y la segunda presionada; duración entre la segunda y la tercera tecla presionada; y duración entre la segunda tecla liberada y la tercera tecla presionada). Para remover los datos atípicos de las pulsaciones agrupadas en dígrafos y trígrafos se calcula la media y desviación estándar

con respecto a los lapsos de tiempo de las pulsaciones, eliminando las combinaciones que contienen desviaciones estándar mayores a la media.

Evaluación

La evaluación se realizó durante dos semanas con 12 usuarios, 10 hombres y dos mujeres, de 24 a 34 años de edad. Los participantes instalaron el software en la computadora que utilizan comúnmente, capturaron textos fijos obtenidos de una obra literaria. Después de haber capturado los datos, se les presentó un cuestionario con 15 estados emocionales a evaluar, en el cual asignaron el nivel percibido de cada emoción después de haber capturado los textos predefinidos. El software permitió clasificar 15 estados emocionales, sin embargo, sólo seis estados presentaron alta precisión con un rango de 77.4% hasta 87.8%, estos estados emocionales fueron: confianza, indecisión, nerviosismo, relajación, tristeza y cansancio.

Conclusiones

En este trabajo de investigación se demuestra que la dinámica del tecleo puede utilizarse para clasificar estados emocionales. Esta investigación aporta un software en el que se incluye la recolección automática de los eventos del teclado, una interfaz gráfica para crear pruebas, utilizando una metodología basada en el muestreo de la experiencia (ESM) y un análisis para evaluar los modelos de clasificación.

3.3 Investigaciones que implementan tecnología relacionada con mouse, teclado e interfaz gráfica de usuario

3.3.1 *LoggerMan: una bitácora y herramienta de visualización para capturar el uso de la computadora [9]*

Descripción general

En esta investigación se desarrolla un software que captura diferentes aspectos de nuestra computadora: variables relacionadas con el teclado, mouse, capturas de pantalla, aplicaciones ejecutadas, portapapeles, así como un módulo que muestra el reporte de la actividad computacional. El software es desarrollado para ser compatible con el sistema operativo Mac OSX 10.7 y versiones posteriores. Los datos que se recolectan en esta aplicación se almacenan localmente en la computadora. LoggerMan también puede ser configurado para ejecutarse después de reiniciar el sistema operativo.

Arquitectura

El software que se implementa incluye diferentes tecnologías, las cuales capturan los eventos detectados registrando la fecha y hora de cada evento, las tecnologías incluidas se integran en diferentes módulos:

1. Módulo relacionado con el teclado. Este módulo es el responsable de capturar los eventos de las pulsaciones del teclado y almacenarlos en archivos locales. Almacena dos tipos de categorías: basado en teclas y basado en palabras al concatenar el flujo de eventos.
2. Módulo relacionado con el mouse. Todas las acciones del mouse son capturadas en este módulo: eventos de los botones del mouse, clics derechos e izquierdos, movimientos hacia arriba, hacia abajo y acciones de arrastrar y soltar.
3. Módulo para capturas de pantalla. Este módulo permite capturar la pantalla de la ventana activa, el intervalo de tiempo de captura que puede seleccionar el usuario es 5, 10 o treinta segundos.
4. Módulo de aplicaciones. Este módulo es diseñado para registrar el nombre de las aplicaciones que se ejecutan, independientemente si se usan o no.
5. Módulo de portapapeles. Este módulo es el responsable de rastrear las operaciones de copiar y pegar. Cualquier texto que es copiado al portapapeles es capturado y almacenado en los archivos locales.

Evaluación

El software se instaló en los equipos de tres participantes durante dos meses, los participantes iniciaron el software presionando doble clic al icono de la aplicación por primera vez. Después, la aplicación se mantuvo trabajando automáticamente en segundo plano. LoggerMan almacenó alrededor de 1GB de datos por semana para cada usuario, el mayor espacio fue utilizado por las capturas de pantalla. El reporte que visualizó el usuario se mostró por medio de gráficas que fueron intuitivas para mostrar el uso de su computadora. Específicamente se graficó el uso del teclado, mouse, aplicaciones y títulos de las capturas de pantalla.

Conclusiones

Este trabajo de investigación aporta un software útil en el dominio de las interacciones humano-computadora. LoggerMan es una herramienta de rastreo y herramienta de visualización para mostrar la actividad computacional y dar pistas basadas en los datos recolectados. LoggerMan puede apoyar a un rango amplio de investigaciones relacionadas con interacciones humano-computadora y ser aplicado para identificación de usuarios, personalidad y cognición.

3.4 Investigaciones que implementan tecnología de comunicación por medio del *Smartphone*

3.4.1 *Smartphone* basado en predicción de estrés [13]

Descripción general

En esta investigación se recolectan automáticamente los datos de sensores para predecir los niveles de estrés que experimenta un usuario. Los datos del *Smartphone* se recolectan utilizando una aplicación que es desarrollada con la interfaz de programación de los servicios de *Google Play*. Los datos de la escala de estrés percibidos se recuperan por medio de una aplicación comercial llamada *movisensXS* dedicada a administrar cuestionarios. El análisis de los datos se lleva a cabo utilizando el programa R, específicamente se utiliza la correlación de Pearson. La predicción se genera por medio del algoritmo de clasificación de regresión lineal incluido en la plataforma de software *WEKA* utilizada comúnmente para el aprendizaje automático.

Arquitectura

La aplicación desarrollada registra el uso del *Smartphone* y datos de sensores. Una vez instalada la aplicación se ejecuta automáticamente en segundo plano, recolectando periódicamente los datos y enviándolos un servidor con soporte PHP y MySQL. La aplicación se divide en diferentes categorías que recuperan cierta información:

1. Actividad. En esta categoría se incluye la actividad física, inmóvil y actividad en la pantalla. En la actividad física se monitorean las actividades de caminar y correr. La parte inmóvil incluye la detección del dispositivo cuando se encuentra en un vehículo o sólo percibe inclinación. La actividad en la pantalla incluye la detección de eventos de encendido y apagado y duración de la sesión incluyendo las sesiones durante las horas de trabajo.
2. Uso de las aplicaciones. El uso de las aplicaciones se contabiliza en 6 categorías de acuerdo con la clasificación de *Google Play*: información, sistema, salud, entretenimiento, social y trabajo.
3. Uso de sensores. Los sensores que se utilizan en esta categoría es el sensor de luz y de sonido. El sensor de luz es utilizado para detectar la cantidad de sesiones iniciadas con luz y en lugares oscuros. El sensor de sonido detecta las medidas máximas, mínimas y medias de la exposición al ruido.
4. Comunicación. En la comunicación se incluye el tráfico de red, mensajes y llamadas. En el tráfico de red se considera la cantidad de tráfico de red en términos de bites, categorizados como recibidos, transmitidos, con la red de datos activa y utilizando la red inalámbrica. En la parte de mensajes se registran los mensajes recibidos, incluyendo los mensajes de los mensajeros de *WhatsApp*, *Google Hangouts* y *Facebook Messenger*. Con relación a las llamadas se detecta cuando una llamada es recibida y se activa el

micrófono para medir los niveles de potencia de voz clasificando también las llamadas recibidas en horarios de trabajo.

Evaluación

La aplicación desarrollada para monitorear el *Smartphone* y la aplicación comercial para aplicar la escala de percepción de estrés fueron instaladas en el dispositivo de 15 participantes durante dos semanas. La escala de estrés percibido se contestó siete veces al día. Los resultados mostraron correlaciones significantes, el ruido presente y la comunicación social mostraron una mayor influencia significativa.

Conclusiones

Este trabajo de investigación aporta una aplicación desarrollada que utiliza la interfaz de programación ofrecida por *Google Play*. Los resultados mostraron correlaciones significativas entre las puntuaciones de las pruebas de detección de estrés y los datos recolectados de manera automática. Con este trabajo se espera que se generen otras investigaciones basadas en el uso de *Smartphone*.

3.4.2 Modelando y descubriendo el comportamiento humano a través del registro de los datos detectados por medio del Smartphone [14]

Descripción general

En esta investigación se implementa un modelo que permite identificar el comportamiento humano argumentado que cada persona presenta un comportamiento único. El modelo procesa información obtenida por medio de múltiples sensores incluidos en un *Smartphone*. Los valores de las variables son obtenidos por medio de una aplicación desarrollada con el *Framework* de programación *Funf* basado en Android, el cual proporciona un conjunto reusable de funcionalidades para obtener un rango amplio de datos de sensores. El modelo implementa medidas de similitud entre cadenas para obtener la precisión de la información procesada.

Arquitectura

La aplicación ha sido desarrollada con el *Framework* de programación *Funf*. Los valores recolectados guardan la fecha y hora en que fueron almacenados en un archivo de texto plano con valores separados por comas. Por cada variable se definió un intervalo de tiempo para recolectar los datos y la duración que equivale al tiempo de grabación de los datos de los sensores. La recolección de datos se clasifica en tres categorías:

1. Petición de datos actuales. En esta categoría se recupera el nombre de las redes inalámbricas cercanas y el nombre de los Bluetooth detectados; el estatus de la batería, que puede ser: descargado, cargado o cargando; el estatus de la pantalla, que puede ser: encendida o apagada; y el nombre de las aplicaciones que se están ejecutando.
2. Datos históricos. En esta categoría se recolectan los números telefónicos que son codificados cuando se almacenan, así como el tipo de mensajes o llamadas.

3. Datos continuos. En esta categoría se incluye el estatus de la actividad basada en los datos del acelerómetro, el cual se clasificada como: ninguna actividad, poca actividad o mucha actividad; y la ubicación con el sensor GPS que se clasifica como: movimiento en la misma ubicación, poco movimiento y mucho movimiento.

El *Framework* de programación *Funf* presenta un problema al recolectar los datos históricos como registro de mensajes y llamadas debido a que los datos se recolectan una vez al día, ocasionando que se generen datos duplicados o algunas veces se retornan valores de error como NA o sin valor. El lenguaje de programación R se utilizó para eliminar los datos duplicados y limpiar el ruido de la información.

Evaluación

La evaluación se llevó a cabo con datos recolectados de 37 estudiantes que participaron durante un mes y 20 días. El tamaño total de los datos que se analizó fue de 28 GB. Los datos recolectados de un mes se utilizaron para construir el modelo mientras que el resto de los datos de 20 días para comprobar su rendimiento en términos de precisión. Para evitar que se agotaran los recursos de la computadora al analizar los 28 GB de datos con el programa R, el proceso de análisis se dividió en tres módulos que almacenaron por separado los resultados en un archivo de texto de valores separados por comas.

El primer módulo se encargó de remover los datos duplicados y de su limpieza, el segundo módulo de la extracción de características y el tercer módulo de transformar los datos al formato correcto para aplicar el modelo de comportamiento, el cual utilizó la distancia de *Levenshtein* para remover espacios entre palabras y medidas similitud para obtener la precisión entre cadenas. La precisión de los datos del modelo se comparó con la de los datos de prueba obteniendo una precisión mayor al 80%, lo cual demostró que el modelo puede ser usado para identificación del usuario.

Conclusiones

Esta investigación aporta una aplicación de recolección de valores de variables obtenidas por medio del *Framework* de programación *Funf* que permite extraer información de múltiples sensores y un modelo basado en medidas de similitud para identificación del usuario. El proceso de creación del modelo implicó una separación de archivos que permitió reducir el tiempo de procesamiento y los recursos de la computadora en donde se analizó el conjunto de información equivalente a 28 GB.

3.5 Conclusiones del estado del arte

Los artículos de investigación que se analizaron anteriormente demostraron que con el uso de las tecnologías: mouse, teclado, interacciones entre ventanas, mensajes y llamadas es posible detectar procesos mentales que se manifiestan a través del comportamiento de las personas que utilizan estas tecnologías o cualquier otra.

En esta investigación se creó una tabla comparativa de las investigaciones mencionadas en el estado del arte. En la Tabla 3.1 se incluyen diferentes criterios que permiten tener una visión más clara de los métodos utilizados en cada investigación para detectar procesos mentales:

- **Investigación.** Menciona el título de la investigación que se analizó.
- **Objetivo.** Menciona la finalidad del trabajo de investigación.
- **Tecnologías.** Menciona los dispositivos físicos utilizados con sensores que envían o reciben señales digitales.
- **Variables.** Se refiere a las propiedades que fueron medidas y cuyos valores cambian constantemente.
- **Software de desarrollo.** Menciona las tecnologías de software utilizadas para desarrollar los sistemas de monitoreo que recabaron la información de las variables.
- **Almacenamiento.** Menciona los medios de almacenamiento que se usaron para recopilar la información que fue recolectada.
- **Sistema operativo.** Se mencionan los sistemas operativos compatibles con el software de monitoreo que fue desarrollado en cada artículo de investigación.
- **Evaluación.** Menciona los tipos de evaluación que se aplicaron para comprobar la funcionalidad de los procesos incluidos en cada artículo con el fin de cumplir su objetivo.

Tabla 3.1 Tabla comparativa de los trabajos relacionados

Investigación	Objetivo	Tecnologías	Variables	Software de desarrollo	Almacenamiento	Sistema Operativo	Evaluación
<p>MouseTracker: Software para el estudio del procesamiento mental incorporando un método computacional que utiliza el rastreo del mouse [10].</p>	<p>Ofrecer un software que permita analizar la percepción de estereotipos asociados y percepción de los sonidos, mediante el análisis de las trayectorias generadas por medio de un mouse computacional.</p>	<p>- Mouse.</p>	<p>- Coordenadas x, y del puntero del mouse.</p>	<p>- Interfaz de programación de Windows.</p>	<p>- Archivo de texto plano separado por comas.</p>	<p>- Windows.</p>	<p>- La precisión del software <i>MouseTracker</i> se obtuvo comparando la información recolectada con el software <i>DirectRT</i>. - El análisis de las trayectorias se generó obteniendo medidas como el área bajo la curva y desviación máxima.</p>
<p>Capturando el interés del usuario a partir del registro de interacciones humano-computadora [12].</p>	<p>Implementar un modelo que permita inferir sobre la utilidad de los documentos web por medio de las interacciones generadas con el mouse.</p>	<p>- Mouse.</p>	<p>- Tiempo de permanencia en las páginas web. - Movimientos del cursor. - Eventos generados con la rueda del mouse. - Texto seleccionado por el mouse.</p>	<p>- Tecnología AJAX.</p>	<p>- Servidor remoto.</p>	<p>- Cualquier sistema operativo con navegador web.</p>	<p>- Leer documentos web personalizados. - Contestar un cuestionario por medio de la escala LIKER con valores de 1 a 5.</p>

Investigación	Objetivo	Tecnologías	Variables	Software de desarrollo	Almacenamiento	Sistema Operativo	Evaluación
Identificación de estados emocionales usando la dinámica de tecleo [11].	Determinar las emociones presentadas por los usuarios analizando el ritmo de sus patrones de escritura sobre un teclado estándar.	- Teclado.	- Teclas independientes (tiempo entre el evento <i>KeyUp</i> y <i>KeyDown</i>). - Teclas agrupadas en dígrafos (tiempo entre el primer y el segundo <i>KeyDown</i> , tiempo entre el primer <i>KeyUp</i> y el segundo <i>KeyDown</i>). - Teclas agrupadas en trígrafos (tiempo entre el primer y segundo <i>KeyDown</i> , tiempo entre el primer <i>KeyUp</i> y segundo <i>KeyDown</i> , tiempo entre el segundo y tercer <i>KeyDown</i> y tiempo entre el segundo <i>KeyUp</i> y tercer <i>KeyDown</i>).	- C#. - Librería <i>Hook</i> de <i>Windows</i> .	- Servidor remoto.	- Windows.	- El software recolectó información de 12 usuarios que capturaron textos fijos durante las pruebas. - El análisis se realizó con base en la evaluación de los datos obtenidos por los modelos generados de cada emoción.
LoggerMan: una bitácora y herramienta de visualización para capturar el uso de la computadora [9].	Interceptar y almacenar eventos del uso de la computadora para un futuro análisis.	- Mouse. - Teclado. - Interfaz gráfica.	- Teclas capturadas. - Palabras capturadas. - Clics derechos. - Clics izquierdos. - Movimiento hacia arriba con la rueda del mouse. - Movimiento hacia abajo con la rueda del mouse. - Eventos de arrastrar y soltar. - Pantalla activa. - Nombre de las aplicaciones que se ejecutan. - Texto copiado al portapapeles.	- Software propio del sistema operativo Mac.	- Archivos de texto plano.	- Mac OSX 10.7 y últimas versiones.	- Instalación del software en los equipos de tres participantes durante dos meses.

Investigación	Objetivo	Tecnologías	Variables	Software de desarrollo	Almacenamiento	Sistema Operativo	Evaluación
<i>Smartphone</i> basado en predicción de estrés [13].	Predecir el nivel de estrés por medio del uso del <i>Smartphone</i> y la información de los sensores incluidos en el dispositivo.	<ul style="list-style-type: none"> - Mensajes. - Llamadas. - Sensor de sonido. - Sensor de luz. - Uso de las aplicaciones. 	<ul style="list-style-type: none"> - Distancia recorrida al caminar. - Distancia recorrida al correr. - Detección en el vehículo. - Detección de inclinación. - Detección de inicio de sesión. - Detección de fin de sesión. - Cantidad de aplicaciones consultadas. - Cantidad máxima, mínima y media de la exposición al ruido. - Cantidad de tráfico de red en términos de bites. - Cantidad de mensajes recibidos. - Niveles de potencia de la voz. 	- Interfaz de programación de Google Play.	- Servidor remoto.	- Android.	<ul style="list-style-type: none"> - Instalación de la aplicación de recolección de datos para el uso del <i>Smartphone</i>. - Instalación de la aplicación comercial para aplicación de la escala de detección de estrés. - Análisis de los datos con correlación de Pearson. - Predicción utilizando el algoritmo de regresión lineal incluido en la plataforma WEKA.
Modelando y descubriendo el comportamiento humano a través del registro de los datos detectados por medio del <i>Smartphone</i> [14].	Construir un modelo de comportamiento humano que pueda ser usado para identificación a partir del registro de valores de	<ul style="list-style-type: none"> - Red inalámbrica. - Bluetooth. - Batería. - Aplicaciones. - Pantalla. - Mensajes. - Llamadas. 	<ul style="list-style-type: none"> - Nombre de las redes inalámbricas más cercanas. - Nombre de los dispositivos Bluetooth más cercanos. - Estatus de la batería. - Estatus de la pantalla. - Nombre de las aplicaciones. - Tipo de mensajes. - Tipo de llamadas. - Números telefónicos de 	- Framework de programación <i>Funf</i> .	- Archivos de texto plano.	- Android.	- Modelo de comportamiento humano con datos recolectados de un mes contra los datos de prueba de 20

Investigación	Objetivo	Tecnologías	Variables	Software de desarrollo	Almacenamiento	Sistema Operativo	Evaluación
	múltiples sensores del <i>Smartphone</i> .	- Acelerómetro. - GPS.	- mensajes y llamadas. - Cantidad de actividad. - Cantidad de movimiento.				días.
Tesis: LaborCheck: Sistema de monitoreo automático de variables usuario-computadora y de interacción social.	Recolectar y almacenar automáticamente información de variables usuario-computadora y de interacción social que pueda ser utilizada para la detección de estrés laboral.	- Mouse. - Teclado. - Ventanas del sistema operativo Windows. - Mensajes. - Llamadas.	- Teclas: total de teclas presionadas, cantidad de teclas tipo retroceso, cantidad de teclas tipo eliminar, cantidad de teclas tipo enter y cantidad de teclas tipo espaciadora. - Movimientos sobre el Mouse: cantidad de clics derechos, cantidad de clics izquierdos, desplazamientos hacia arriba, desplazamientos hacia abajo y pixeles recorridos. - Interfaz gráfica: cantidad de ventanas abiertas y cantidad de cambios entre ventanas. - Cantidad de mensajes: recibidos laborales, enviados laborales, recibidos personales y enviados personales. - Cantidad de llamadas: recibidas laborales, realizadas laborales, perdidas laborales, recibidas personales, realizadas personales y perdidas personales.	- C#. - Librería Hook. - Java sobre Android. - PHP.	- Base de datos remota con MySQL. - Base de datos local con SQLite.	- Windows. - Android.	- Pruebas funcionales del software desarrollado. - Evaluación de la precisión del software desarrollado contra otro software.

Capítulo 4

Sistema LaborCheck

Capítulo 4. Sistema LaborCheck

En este capítulo se describe el sistema de monitoreo LaborCheck. Los aspectos de análisis, diseño y desarrollo se detallan dentro de las secciones que describen cada una de las aplicaciones del sistema. También se mencionan las tecnologías relacionadas al monitoreo automático y de manera técnica las librerías que se implementaron y los métodos que se programaron en las aplicaciones del sistema.

4.1 Metodología de solución

La metodología de solución se puede observar en la Figura 4.1, la cual muestra el conjunto de procesos de cada fase, así como la relación entre los procesos y la base de datos que concentra la información de las tecnologías de monitoreo: teclado, mouse, ventanas, mensajes y llamadas.

La fase 1 integra tres aplicaciones que conforman el sistema LaborCheck: aplicación de interacción usuario-computadora, aplicación de interacción social y aplicación web. La fase 2 integra las pruebas funcionales basadas en el estándar IEEE 829-2008 y las pruebas generadas con otro software. El contenido de la fase 1 se detalla en este capítulo y el contenido de la fase 2 está incluido en el capítulo 5 de pruebas y resultados.

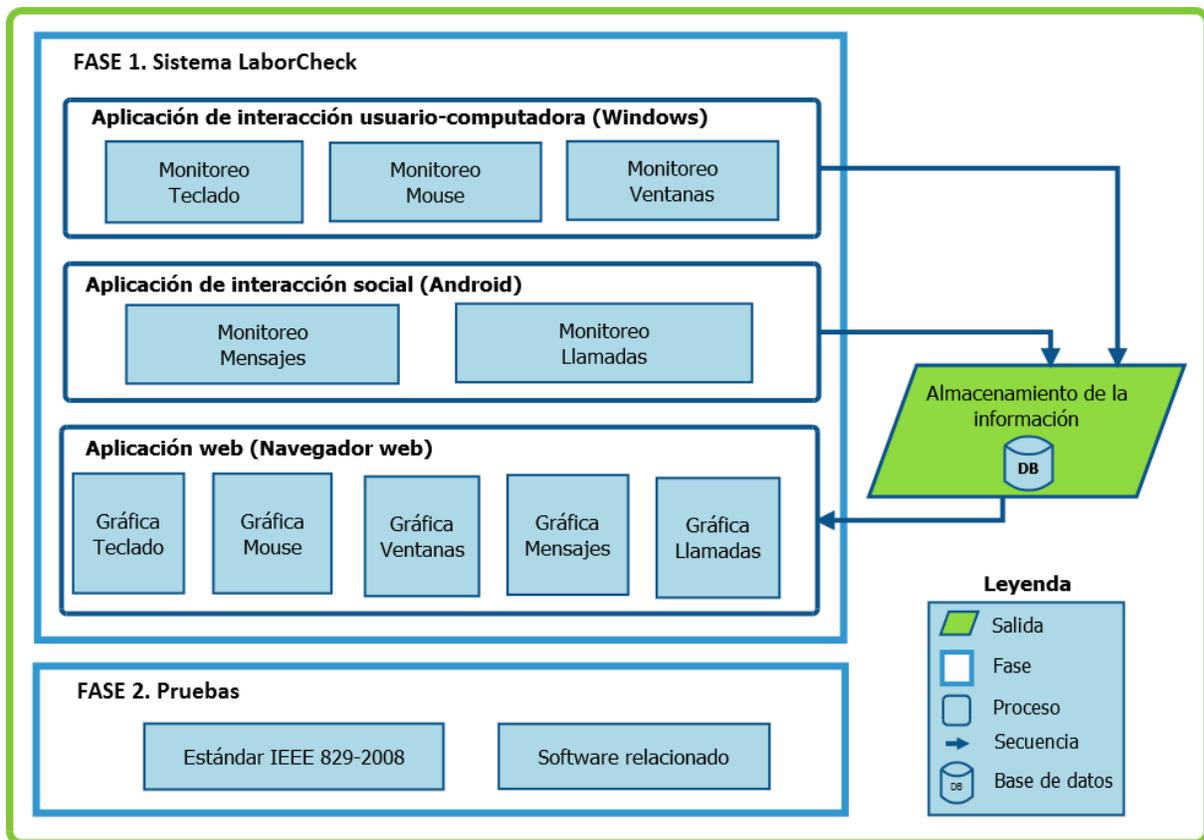


Figura 4.1 Metodología de solución

Fase 1. Sistema LaborCheck: En esta fase se integran las aplicaciones del sistema de monitoreo con los procesos que forman parte de cada aplicación:

Aplicación de interacción usuario-computadora. Esta aplicación está integrada de tres procesos que monitorean las siguientes tecnologías: teclado, mouse y ventanas. Cada proceso recolecta automáticamente las interacciones generadas por los usuarios que utilizan las tecnologías de esta aplicación:

- **Monitoreo teclado.** En este proceso se monitorean automáticamente las interacciones que generan los usuarios por medio del teclado, se detecta la cantidad de pulsaciones que se capturan sobre los programas que se ejecutan en el sistema operativo Windows. También se consideran los mecanismos que permiten interceptar los datos de entrada proporcionados por el teclado para ser contabilizados y almacenados por la aplicación de interacción usuario-computadora.
- **Monitoreo mouse.** En este proceso se monitorean automáticamente las interacciones que generan los usuarios con el uso del mouse, se detectan los movimientos del mouse generados dentro del sistema operativo Windows. Además, se consideran los mecanismos que permiten interceptar los movimientos generados por el mouse para ser contabilizados y almacenados por la aplicación de interacción usuario-computadora.
- **Monitoreo ventanas.** En este proceso se monitorean automáticamente las interacciones que generan los usuarios con las ventanas, estas ventanas se crean al abrir carpetas y cualquier programa instalado en el sistema operativo Windows. También se consideran los mecanismos que permiten interceptar los cambios generados con las ventanas, estos cambios son contabilizados y almacenados localmente por la aplicación de interacción usuario-computadora.

Aplicación de interacción social. Esta aplicación integra dos procesos que recolectan y almacenan automáticamente los valores de las variables de interacción social. Las interacciones capturadas en esta aplicación son detectadas a través de las tecnologías de comunicación de mensajes y llamadas:

- **Monitoreo mensajes.** En este proceso se monitorea automáticamente la interacción social que genera el usuario y sus contactos por medio de los mensajes. Además, se consideran los métodos que permiten detectar los cambios generados en el registro de mensajes, los cuales son contabilizados y almacenados localmente por la aplicación de interacción social.
- **Monitoreo llamadas.** En este proceso se monitorea automáticamente la interacción que genera el usuario y sus contactos por medio de llamadas telefónicas. Asimismo, se consideran los métodos que permiten detectar los cambios generados en la carpeta de registro de llamadas, los cuales se contabilizan inmediatamente cuando se detecta un cambio, el conteo de llamadas se almacena localmente por la aplicación de interacción social.

Como se mencionó anteriormente, la aplicación de interacción usuario-computadora y la aplicación de interacción social almacenan los valores de las variables localmente. Sin embargo, cada uno de los procesos envían los valores de las variables almacenados localmente a la base de datos remota que concentra la información de las tecnologías que son monitoreadas: teclado, mouse, ventanas, mensajes y llamadas.

Aplicación web. Esta aplicación integra cinco procesos que permiten visualizar gráficamente la información de las tecnologías que son monitoreadas:

- Gráfica teclado. En este proceso se obtiene la información recolectada de las interacciones generadas con el teclado. Además, se generan las peticiones hacia las tablas que contienen los valores de la información relacionada con esta tecnología.
- Gráfica mouse. En este proceso se obtiene la información recolectada de las interacciones detectadas por medio del mouse. La información es recuperada después de generar peticiones hacia las tablas que contienen los valores de la información relacionada con esta tecnología.
- Gráfica ventanas. En este proceso se obtiene la información de las interacciones generadas entre las ventanas del sistema operativo Windows. Esta información es mostrada gráficamente con los valores recuperados de las consultas generadas sobre las tablas que contienen esta información.
- Gráfica mensajes. En este proceso se obtiene la información de la interacción social generada por medio de la cantidad de mensajes de texto. Esta información es recuperada al consultar las tablas que almacenan los valores de esta tecnología.
- Gráfica llamadas. En este proceso se obtienen los valores de las variables relacionadas con la interacción generada por medio de la cantidad de llamadas telefónicas. Esta información se recupera al generar peticiones a las tablas que almacenan la información relacionada con llamadas.

El tipo de gráficas en las que se despliega la información recuperada son gráficas de series con diferentes intervalos de tiempo. Para las tecnologías de teclado, mouse y ventanas el intervalo de tiempo que se considera es cada cinco minutos y para las tecnologías de mensajes y llamadas es de dos horas.

Fase 2. Pruebas: En esta fase se integran los procesos que permiten evaluar el sistema de monitoreo, la evaluación se realiza de acuerdo a dos procesos:

- Estándar IEEE 829-2008. En este proceso se comprueba la funcionalidad del sistema, en estas pruebas se supervisan los datos de entrada y se comprueba que las salidas sean correctas.
- Software relacionado. En este proceso se compara la precisión de las interacciones recuperadas por el sistema LaborCheck contra otro software que recupera las interacciones que están relacionadas directamente con las tecnologías incluidas en esta investigación.

Cada aplicación que forma parte del sistema LaborCheck incorpora su propio plan de pruebas debido a que se ejecutan en diferentes plataformas de software. La aplicación de interacción usuario-computadora se ejecuta con el sistema operativo Windows, la aplicación de interacción social con el sistema operativo Android mientras que la aplicación web con un navegador instalado en cualquier sistema operativo.

4.2 Esquema general del sistema LaborCheck

El esquema general de la Figura 4.2 muestra los componentes del sistema. Estos componentes permiten que sean monitoreadas automáticamente las interacciones que generan los usuarios por medio de dos aplicaciones de monitoreo: aplicación de interacción usuario-computadora y aplicación de interacción social. La primera monitorea el uso de la computadora con sistema operativo Windows y la segunda la comunicación del *Smartphone* con sistema operativo Android. La información recolectada por este sistema puede ser consultada por medio de una aplicación web a través de un navegador como Google Chrome, Internet Explorer o Mozilla Firefox.

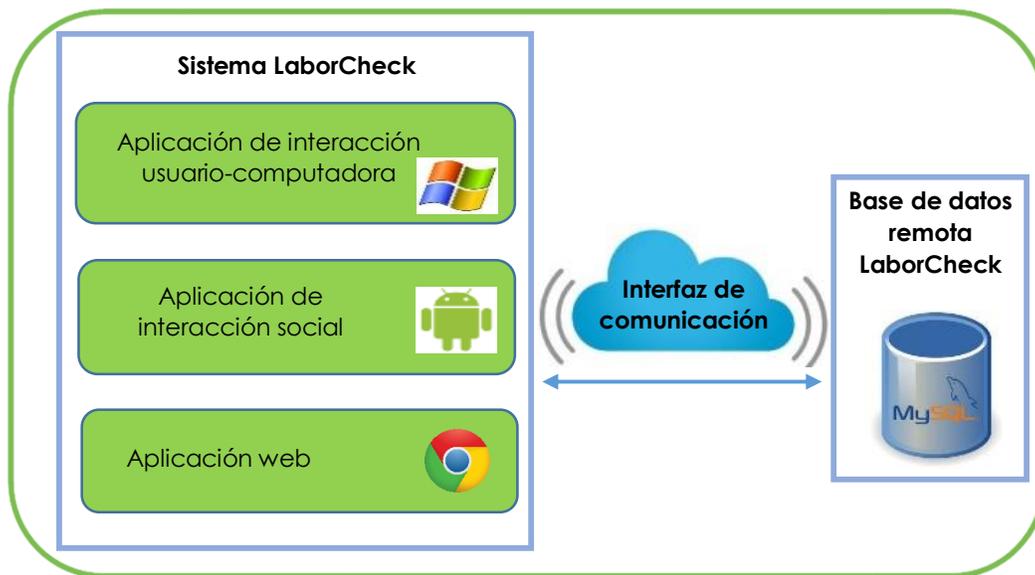


Figura 4.2 Esquema general del sistema LaborCheck

El sistema LaborCheck es un sistema distribuido que permite la interoperabilidad entre las aplicaciones, está basado en el modelo cliente-servidor. Cada aplicación actúa como cliente, las cuales generan peticiones hacia el servidor en donde se encuentra alojada la base de datos remota. El uso de Hardware (mouse y teclado) e interfaz gráfica (interacciones entre ventanas) se monitorea en la aplicación de interacciones usuario-computadora. El uso de las tecnologías de comunicación (mensajes y llamadas) se monitorea por medio de la aplicación de interacción social. La interfaz de comunicación es el medio por el cual se comunican las aplicaciones con la base de datos remota para poder gestionar diferentes operaciones de consulta e inserción. El sistema gestor de base de datos MySQL es el medio por el cual se almacena la información obtenida por las aplicaciones de monitoreo. Con respecto a la Pagina web, esta es desarrollada de tal manera que se adapta al tamaño de la pantalla de los diferentes dispositivos que existen actualmente en el mercado. Por lo tanto, las gráficas que presentan la información se visualizan correctamente en cualquier computadora y dispositivo móvil.

4.3 Aplicación de interacción usuario-computadora

Esta aplicación recolecta automáticamente las interacciones usuario-computadora, las almacena localmente y posteriormente las envía al servidor remoto. Las etapas incluidas en el desarrollo de software están basadas en el modelo en cascada. Dentro del análisis se incluyen los requerimientos funcionales junto con el diagrama de casos de uso, así como los requerimientos no funcionales. En la etapa de diseño se incluye la arquitectura y la base de datos. En la implementación se mencionan los mecanismos que permiten programar la aplicación de interacción usuario-computadora.

4.3.1 Análisis de la aplicación de interacción usuario-computadora

Los requerimientos funcionales para la aplicación de interacción usuario-computadora se describen en la Tabla 4.1. La primera columna describe la funcionalidad de la aplicación y las columnas de almacenamiento determinan si la funcionalidad interactúa con la base de datos local, con la base de datos remota o con ambas.

Tabla 4.1 Requerimientos funcionales de la aplicación de interacción usuario-computadora

Funcionalidad	Almacenamiento local	Almacenamiento remoto
La aplicación permite ingresar los datos de registro para una nueva cuenta de usuario.	X	X
La aplicación permite que el usuario inicie sesión por primera vez con su cuenta existente.	X	X
La aplicación inicia automáticamente después de haber reiniciado el equipo.	X	
La aplicación detecta automáticamente las interacciones generadas sobre el teclado.	X	
La aplicación detecta automáticamente las interacciones generadas sobre el mouse.	X	
La aplicación detecta automáticamente las interacciones generadas entre las ventanas del sistema operativo Windows.	X	
Las interacciones usuario-computadora son almacenadas localmente cada 5 minutos.	X	
Las interacciones usuario-computadora son enviadas al servidor remoto cada 17 minutos.	X	X

Los requerimientos funcionales se describen gráficamente en la Figura 4.3. La especificación general de los casos de uso es descrita y comprobada directamente en el capítulo 5 de pruebas de resultados. Por lo tanto, en este apartado sólo se muestra el diagrama de casos de uso.

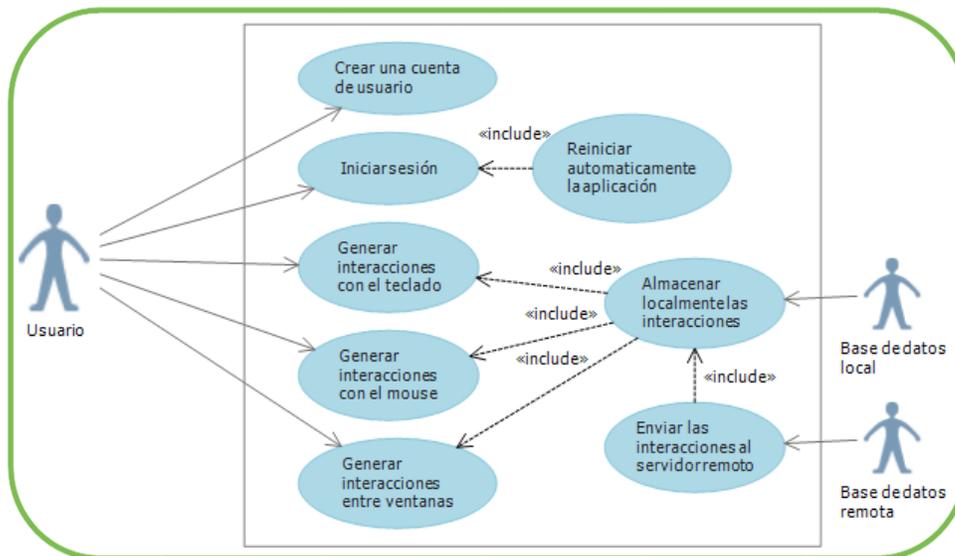


Figura 4.3 Diagrama de casos de uso de la aplicación de interacción usuario-computadora

Los requerimientos no funcionales se muestra en la Tabla 4.2 , los cuales se clasifican de acuerdo a tres categorías: hardware compatible por la aplicación, especificaciones de software y aspectos de confidencialidad.

Tabla 4.2 Requerimientos no funcionales de la aplicación de interacción usuario-computadora

Clasificación	Requerimiento
Hardware	Las interacciones del teclado son recuperadas por cualquier modelo de teclado.
	Las interacciones del mouse son recuperadas por cualquier modelo de mouse.
Software	Las interacciones usuario-computadora son recolectadas bajo el sistema operativo Windows sin importar en que programa se encuentre trabajando el usuario.
	La interfaz gráfica del usuario es diseñada de forma no intrusiva, por lo tanto permite que el usuario se mantenga trabajando sin problemas de interrupción.
	La aplicación de interacciones usuario-computadora es compatible con las versiones de Windows más actuales: Windows 7, Windows 8 y Windows 10.
Confidencialidad	Sólo se almacena la cantidad de interacciones generadas por el teclado, ninguna cadena de caracteres que permita obtener información confidencial es almacenada.
	Respecto al mouse sólo se almacena cantidad de movimientos, no se almacena ninguna información que sea seleccionada por el cursor del mouse.
	Sólo el usuario conoce sus datos de registro: contraseña y nombre de usuario.

4.3.2 Diseño de la aplicación de interacción usuario-computadora

Arquitectura. El diseño de la arquitectura en el que se basa la aplicación de interacción usuario-computadora se puede observar en la Figura 4.4. En esta arquitectura se muestran específicamente los componentes que integran la aplicación y la información que se recupera por la aplicación. Dentro de ellos se encuentran los módulos gráficos de registro e inicio de sesión, los componentes que detectan las interacciones usuario-computadora, el proceso que almacena las interacciones en la base de datos local y el proceso que envía la información al servidor remoto. El servidor remoto concentra todos los tipos de interacción incluidos en el sistema LaborCheck.

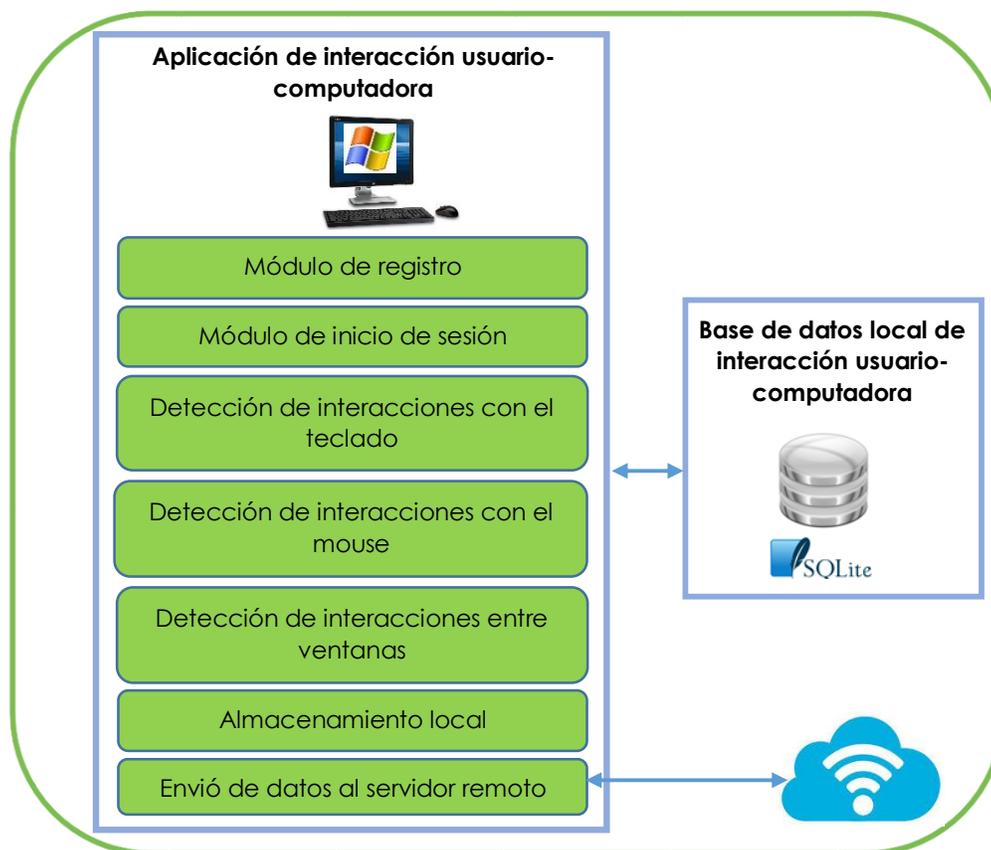


Figura 4.4 Arquitectura de la aplicación de interacción usuario-computadora

Módulo de registro. Este módulo permite capturar los datos de registro del usuario. En este caso sólo permite ingresar un nombre de usuario que no revele la identidad del usuario y una contraseña que sólo el usuario debe conocer. Los datos ingresados son almacenados en la base de datos remota y el *id* insertado se recupera por la aplicación y es almacenado en la base de datos local. La aplicación se reinicia posteriormente y automáticamente recupera este *id* como dato de autenticación.

Módulo de inicio de sesión. Este módulo permite ingresar el nombre de usuario y contraseña en caso de que el usuario no este ejecutando la aplicación. Los datos ingresados son consultados en la base de datos remota para verificar que el usuario exista. Posteriormente el *id* recuperado de la base de datos remota es almacenado en la base de datos local para ser detectado posteriormente por el proceso que inicia la aplicación automáticamente después de reiniciar el equipo.

Detección de interacciones con el teclado. En este módulo se recuperan las interacciones que genera el usuario con el teclado, específicamente se recuperan los valores de las variables que se muestran en columna de la tecnología teclado mostrada en la Tabla 4.3. Estos valores son almacenados temporalmente en la base de datos local y posteriormente se envían a la base de datos remota.

Detección de interacciones con el mouse. En este módulo se recuperan las interacciones que genera el usuario con el mouse, los valores de las variables que se recuperan se observan en la Tabla 4.3. Estos valores se almacenan temporalmente en la base de datos local y posteriormente se envían a la base de datos remota.

Detección de interacciones entre ventanas. En este módulo se recuperan las interacciones que el usuario genera al interactuar con las ventanas de Windows. La especificación de las variables relacionadas con la tecnología ventanas se puede observar en la Tabla 4.3. Estos valores se envían temporalmente a la base de datos local y posteriormente se envían a la base de datos remota.

Almacenamiento local. Este módulo recupera la cantidad de interacciones almacenadas en memoria cada cinco minutos. Los valores de las variables son reiniciados a cero después de este tiempo y los nuevos valores se vuelven a contabilizar para volver a ser almacenados de manera temporal en la base de datos local.

Envío de datos al servidor. Los valores de las interacciones recuperadas por el teclado, mouse y ventanas de Windows son recuperados de la base de datos local y enviados a la base de datos remota en donde se concentra la información del sistema LaborCheck.

Tabla 4.3 Variables de interacción usuario-computadora

Tecnología	Variables
Teclado	Cantidad total de teclas presionadas
	Cantidad de teclas tipo retroceso
	Cantidad de teclas tipo eliminar
	Cantidad de teclas tipo enter
	Cantidad de teclas tipo espaciadora
Mouse	Cantidad total de clics
	Cantidad de clics izquierdos
	Cantidad de clics derechos
	Cantidad de desplazamientos generados hacia arriba con la rueda del mouse
	Cantidad de desplazamientos generados hacia abajo con la rueda del mouse
	Cantidad de pixeles recorridos
Ventanas de Windows	Cantidad de ventanas abiertas
	Cantidad de cabios entre ventanas

Base de datos local. El diseño de la base de datos local se muestra en la Figura 4.5. Esta base de datos se almacena físicamente en el disco duro de la computadora, la base de datos utiliza el gestor de base de datos SQLite. La ruta en la que se ubica es en la carpeta usuarios del sistema operativo Windows. La descripción de las tablas se menciona brevemente en el anexo A.

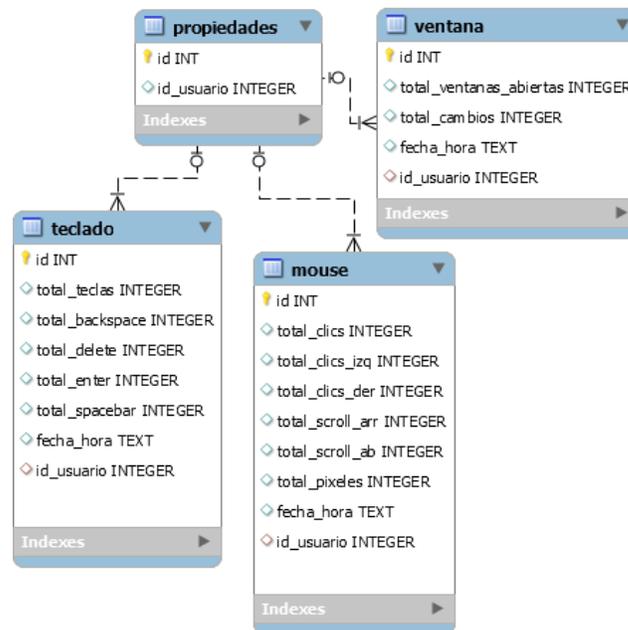


Figura 4.5 Diagrama entidad relación de la base de datos local para la aplicación de interacción usuario-computadora

4.3.3 Implementación de la aplicación de interacción usuario-computadora

En esta sección se describe la implementación de la aplicación de interacción usuario-computadora, en la cual se mencionan las herramientas de desarrollo de software, las librerías, las clases, los métodos y el modelo de programación. La versión final de la aplicación se puede observar en la Figura 4.6., en la cual se muestra la aplicación con una interfaz oculta al usuario, la cual permite que el usuario continúe trabajando con su computadora de manera normal.

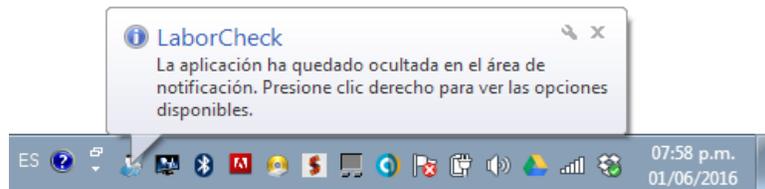


Figura 4.6 Aplicación de interacción usuario-computadora

El entorno de desarrollo integrado que se utilizó para programar la aplicación de interacción usuario-computadora es Microsoft Visual Studio con el lenguaje de programación C Sharp (C#). La programación de esta aplicación está basada en el modelo orientado a objetos. Los principales métodos desarrollados para la aplicación de interacciones usuario-computadora se describen en la Tabla 4.4. Esta tabla se categoriza de acuerdo a los módulos incluidos en la arquitectura presentada en la etapa de diseño, cada módulo está incluido en los diferentes formularios que se muestran gráficamente al usuario.

Se desarrollaron tres formularios para esta aplicación. El primer formulario permite registrar a un nuevo usuario, el cual será monitoreado. El segundo formulario de inicio de sesión permite autenticar al usuario una vez que ya está registrado y el tercer formulario es el que se minimiza en forma de un icono sobre el área de notificación de la barra de tareas del sistema operativo Windows y se encuentra monitoreando las interacciones usuario-computadora.

Los módulos de registro e inicio de sesión se implementan en dos formularios que comparten la clase usuario, esta clase permite crear una instancia para validar los datos del usuario.

Las interacciones usuario-computadora se implementan dentro del formulario de monitoreo. Este formulario utiliza el mecanismo *hook* proporcionado por *Microsoft Windows* para interceptar los eventos del sistema. Con base en este mecanismo se desarrollaron las clases con los métodos necesarios para interceptar los eventos de las tecnologías incluidas en esta investigación: teclado, mouse y ventanas.

El módulo de almacenamiento local también se incluye dentro del formulario de monitoreo. Dentro de él se incluyen diferentes métodos que se ejecutan en segundo plano cada cinco minutos y que almacenan los valores de las variables de las interacciones en la base de datos local gestionada con SQLite.

El módulo de envío de información opera en segundo plano en el formulario de monitoreo. Las interacciones se recuperan de la base de datos local sobre cadenas de datos con formato

JSON. Estos valores se envían al servidor a través de una interfaz cliente incluida en la aplicación.

Tabla 4.4 Métodos programados para la aplicación de interacción usuario-computadora

Módulo	Formulario	Clase	Método	Descripción
Registro	Registro	Usuario	Validar Datos	En este método se valida que el nombre de usuario no exista en la base de datos remota.
Inicio de sesión	Iniciar Sesión	Usuario	Validar Usuario	En este método se valida que el usuario y la contraseña correspondan con el usuario almacenado en la base de datos remota.
Detección de interacciones con el teclado	Monitoreo	GlobalKey BoardHook	gHook KeyDown	En este método se detectan y contabilizan los diferentes eventos que se generan con el teclado, se detectan los tipos de teclas considerados en esta investigación: retroceso, enter, eliminar y barra espaciadora.
Detección de interacciones con el mouse	Monitoreo	GlobalMouse Hook	gHook MouseDown	En este método se detectan y contabilizan los eventos generados con el botón derecho e izquierdo.
			gHook MouseWheel	En este método se detectan y contabilizan los eventos generados con la rueda del mouse: cantidad de ciclos generados hacia arriba y cantidad de ciclos generados hacia abajo.
			gHook MouseMove	En este método se detectan y contabilizan los eventos generados con el puntero del mouse.
Detección de interacciones entre ventanas	Monitoreo	GlobalWindow Hook	Listener SystemEvent	En este método se detectan y contabilizan los eventos que corresponden a las ventanas ejecutadas en primer plano.
Almacenamiento local	Monitoreo	Background Worker	DoWork Teclado	En este método se recuperan cada cinco minutos los valores de las variables que contabilizan los eventos del teclado y se almacenan en la tabla teclado de la base de datos local.
			DoWork Mouse	En este método se recuperan cada cinco minutos los valores de las variables que contabilizan los eventos del mouse y se almacenan localmente
			DoWork Ventana	En este método se recuperan cada cinco minutos los valores de las variables que contabilizan los eventos de las ventanas y se almacenan en la base de datos local.
Envío de datos al servidor	Monitoreo	Background Worker	DoWork Envía	En este método se recuperan cada 17 minutos todas las interacciones del teclado, mouse y ventanas que se encuentran en la base de datos local y en seguida se envían a la base de datos remota. Después de que se envían correctamente se eliminan de la base de datos local.

4.4 Aplicación de interacción social

Esta aplicación recolecta automáticamente las interacciones asociadas a la comunicación social que genera un usuario con sus contactos. Estas interacciones se almacenan localmente en el dispositivo móvil y posteriormente se envían a la base de datos que se encuentra alojada en el servidor remoto. Las etapas de desarrollo de software están centradas sobre el modelo de desarrollo en cascada. En la etapa de requerimientos se incluyen los requerimientos funcionales y los requerimientos no funcionales. En la etapa de diseño se encuentra la arquitectura de la aplicación y el diseño de la base de datos local y en la etapa de implementación se incluyen los detalles técnicos para programar la aplicación de interacción social.

4.4.1 Análisis de la aplicación de interacción social

Los requerimientos funcionales para la aplicación de interacción social están descritos en la Tabla 4.5. La primera columna describe la funcionalidad que cumple la aplicación, la segunda está marcada cuando la funcionalidad interactúa con la base de datos local y la tercera columna esta marca cuando la funcionalidad interactúa con la base de datos remota. Sin embargo, algunas funcionalidades interactúan con los dos medios de almacenamiento.

Tabla 4.5 Requerimientos funcionales de la aplicación de interacción social

Funcionalidad	Almacenamiento local	Almacenamiento remoto
La aplicación permite que el usuario inicie sesión con los datos de registro ingresados en la aplicación de interacción usuario-computadora.	X	X
La aplicación inicia automáticamente después de haber reiniciado el dispositivo móvil.	X	
La aplicación permite registrar localmente contactos para clasificarlos como personales y laborales.	X	
La aplicación permite realizar cambios sobre los contactos agregados.	X	
La aplicación permite eliminar los contactos para omitirlos en el monitoreo.	X	
La aplicación detecta automáticamente la interacción entre el usuario y sus contactos por medio de los mensajes.	X	
La aplicación detecta automáticamente la interacción entre el usuario y sus contactos por medio de las llamadas.	X	
Las interacciones de comunicación social se almacenan localmente cada dos horas.	X	
Las interacciones de comunicación social se envían al servidor remoto cada dos horas y media.	X	X

Los requerimientos funcionales se muestran gráficamente en la Figura 4.7. La documentación que especifica los requerimientos de los casos de uso se describe con más detalle en el capítulo 5 dentro del plan de las pruebas funcionales para la aplicación de interacción social. Por lo tanto, en esta sección sólo se muestra el diagrama en donde se aprecia gráficamente la relación entre los casos de uso y los actores que interactúan en la aplicación de interacción social.

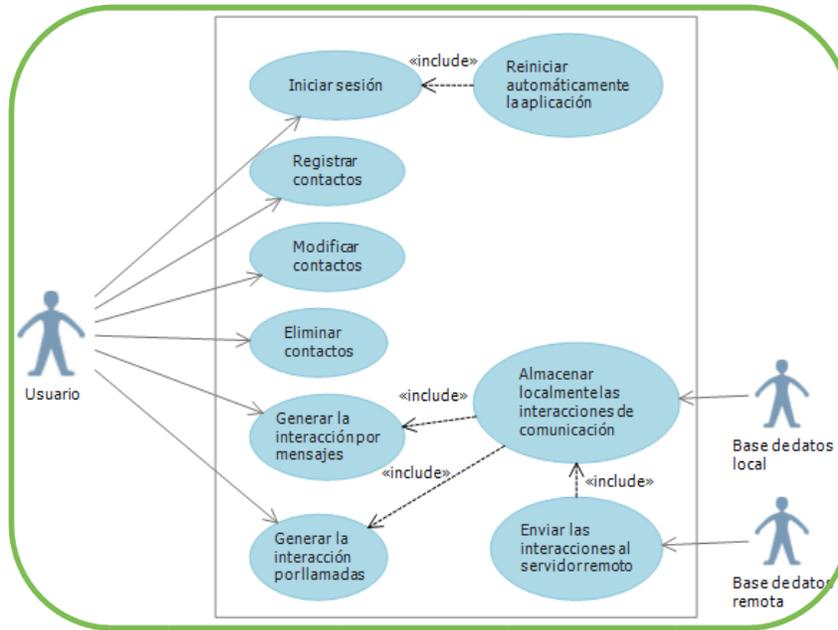


Figura 4.7 Diagrama de casos de uso de la aplicación de interacción social

Los requerimientos no funcionales se describen en la Tabla 4.6. Estos requerimientos se clasifican de acuerdo en dos categorías: software compatible con la aplicación y requerimientos de confidencialidad.

Tabla 4.6 Requerimientos no funcionales de la aplicación de interacción social

Clasificación	Requerimiento
Software	Las interacciones de comunicación social son recuperadas por medio de un dispositivo móvil con sistema operativo Android.
	La interfaz gráfica de la aplicación muestra una interfaz que es fácil de usar para administrar los contactos.
	La aplicación es compatible con las versiones más actuales de Android: Android 4.1, Android 5.0 y Android 6.0.
	La aplicación agrega la información de los contactos de la agenda para agilizar el ingreso de los datos del contacto.
Confidencialidad	En el servidor remoto sólo se almacena la cantidad de mensajes detectados, no se

Clasificación	Requerimiento
	almacena ningún carácter o palabra contenida en el mensaje.
	Respecto a las llamadas, se almacena sólo la cantidad de llamadas detectadas, no se recupera ninguna pista de audio que revele el contenido de la llamada.
	La información de contactos como nombre y número telefónico sólo se almacena localmente en el dispositivo móvil del usuario debido a que es información confidencial. Estos contactos se almacenan localmente para clasificar la cantidad de interacciones de comunicación generadas entre el usuario y sus contactos.
	El usuario tiene acceso a la información recuperada por medio de un enlace que puede ver en la pantalla de bienvenida de la aplicación de interacción social.

4.4.1 Diseño de la aplicación de interacción social

Arquitectura de la aplicación. El diseño de la arquitectura que recupera las interacciones de comunicación generadas por medio de mensajes y llamadas se muestra en la Figura 4.8. En esta arquitectura se integran los componentes que permiten clasificar los contactos como personales o laborales. También se integran los componentes que contabilizan automáticamente las interacciones generadas con los mensajes y llamadas, el proceso que almacena los valores detectados en la base de datos local y el proceso que envía estos valores a la base de datos remota.



Figura 4.8 Arquitectura de la aplicación de interacción social

Módulo de inicio de sesión. Este módulo permite que se ingresen los datos de autenticación: nombre de usuario y contraseña. Estos datos se validan en la base de datos remota para verificar que el usuario este registrado. Después de la validación, estos datos son almacenados en la base de datos local para que la próxima vez que se reinicie el dispositivo móvil, la aplicación se reinicie automáticamente tomando estos datos como autenticación de usuario.

Módulo de contactos. En este módulo se registran los contactos del usuario y se clasifican como personales o laborales, lo cual permite que la aplicación pueda clasificar los mensajes y llamadas cuando son detectados. En este módulo también se puede modificar la categoría de los contactos o eliminarlos para que no sean tomados en cuenta por esta aplicación.

Detección de mensajes. En este módulo se detectan automáticamente los mensajes que se registran en el dispositivo móvil. Cada vez que llega un mensaje se incrementa el valor almacenado en la base de datos. Las variables que se detectan se muestran en la Tabla 4.7.

Detección de llamadas. En este módulo se detectan automáticamente las llamadas que se registran en el dispositivo móvil. Cada vez que se registra una llamada en el registro del teléfono es contabilizada por la aplicación de monitoreo y el valor almacenado en la base de datos se incrementa. Las variables que se detectan en este módulo se pueden observar en la Tabla 4.7.

Almacenamiento local. Este módulo permite que los valores acumulados de las llamadas y mensajes detectados se recuperen cada dos horas y se almacenen en las tablas que registran estas interacciones con el mismo lapso de tiempo.

Envío de datos al servidor remoto. Los valores de las variables que se encuentran en las tablas que almacenan los datos cada dos horas se envían a la base de datos remota en donde se concentran los valores de todas las interacciones que se consideran en el sistema LaborCheck.

Tabla 4.7 Variables de interacción social

Tecnología	Variables
Mensajes	Cantidad de mensajes laborales recibidos
	Cantidad de mensajes personales recibidos
	Cantidad de mensajes laborales enviados
	Cantidad de mensajes personales enviados
Llamadas	Cantidad de llamadas laborales recibidas
	Cantidad de llamadas personales recibidas
	Cantidad de llamadas laborales realizadas
	Cantidad de llamadas personales realizadas
	Cantidad de llamadas laborales perdidas
	Cantidad de llamadas personales perdidas

Base de datos local. El modelo relacional de la base de datos para la aplicación de interacción social se muestra en la Figura 4.9. Esta base de datos se almacena físicamente en el dispositivo móvil utilizando el sistema gestor de base de datos SQLite.

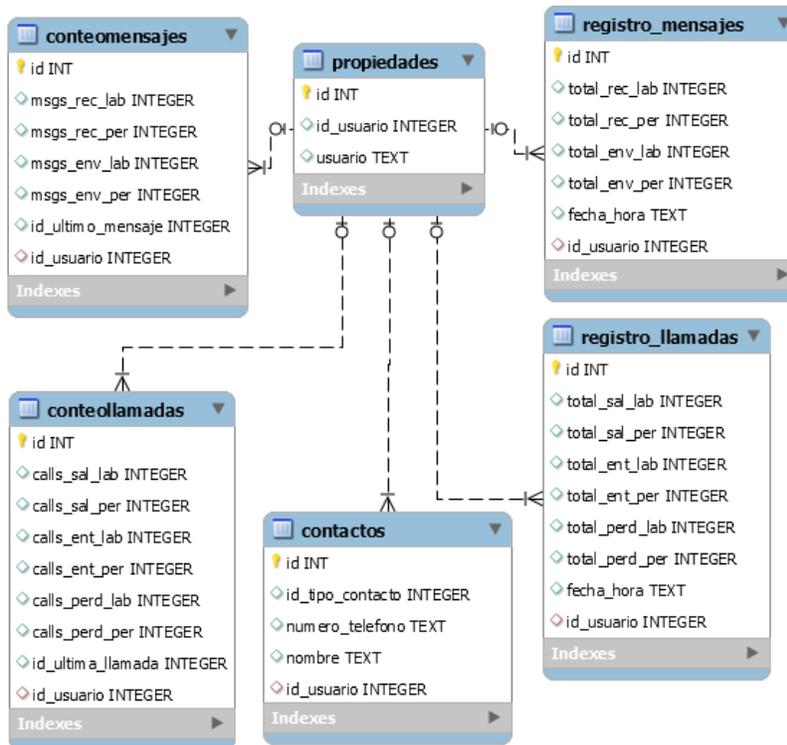


Figura 4.9 Diagrama entidad relación de la base de datos local para la a aplicación de interacción social

Las tablas que se muestran en el modelo entidad relación permiten almacenar las interacciones generadas entre el usuario y sus contactos, en el anexo B se describen brevemente las tablas y los campos de cada tabla.

4.4.3 Implementación de la aplicación de interacción social

En esta sección se describe la implementación de la aplicación de interacción social, se mencionan las herramientas de desarrollo de software, librerías, clases, métodos y modelos de programación que se utilizaron en el desarrollo de esta aplicación. La interfaz gráfica de la aplicación se puede observar en la Figura 4.10. Esta interfaz muestra al usuario un enlace a los datos que están siendo monitoreados. En el capítulo 5 de pruebas y resultados se puede observar el resto de las interfaces gráficas programadas con las herramientas descritas en esta sección.



Figura 4.10 Aplicación de interacción social

La aplicación de interacción social se programó sobre el entorno de desarrollo integrado *Android Studio*, el cual trabaja con el modelo orientado a objetos. En este apartado se integran los principales métodos que pertenecen a las clases que se programaron en esta aplicación, clasificados de acuerdo al módulo al que pertenecen, los cuales están descritos en la Tabla 4.8.

Esta aplicación cuenta dos tipos de interfaces: interfaces gráficas que se utilizan en el módulo de inicio de sesión y en el módulo de contactos e interfaces programadas por medio de servicios que se ejecutan en segundo plano para la detección de mensajes, llamadas, almacenamiento local y almacenamiento remoto.

Todos los servicios programados se ejecutan automáticamente al iniciar la aplicación. También se ejecutan después de reiniciar el dispositivo móvil al disponer de la clase *BroadcastReceiver* que permite detectar los eventos generados por el sistema operativo Android.

Para registrar las llamadas y los mensajes cada dos horas se utilizó la clase *AlarmManager*. De la misma manera esta clase se utilizó para enviar los valores registrados de la base de datos local hacia la base de datos remota cada dos horas y media.

Tabla 4.8 Métodos programados para la aplicación de interacción social

Módulo	Interfaz	Clase	Método	Descripción
Inicio de sesión	Activity_main	MainActivity	TareaWS	En este método se valida que el usuario y la contraseña coincidan con uno de los usuarios registrados en la base de datos remota y se obtiene el <i>id</i> para ser almacenado localmente.
Contactos	Fragment_contactos	Contactos fragment	TareaInsertar ContactoWS	En este método se insertan los datos del nuevo contacto, antes de ser insertado se valida que el número de teléfono sea válido.
			TareaActualizar ContactoWS	En este método se actualiza el nombre y número del contacto, antes de actualizar se valida que el número de teléfono sea diferente a los que existen en la base de datos local.
			TareaEliminar ContactoWS	Este método permite eliminar el contacto de la base de datos local.
Detección de mensajes	Phone Service	SMSObserver	contabilizarMensaje	Este método se ejecuta después de detectar cambios en el registro de mensajes e incrementa el valor del tipo de mensaje y contacto de la base de datos local.
Detección de llamadas	Phone Service	CALLS Observer	Contabilizar Llamadas	Este método se ejecuta después de detectar cambios en el registro de llamadas y posteriormente incrementa el valor de los campos de acuerdo al tipo de llamada y contacto en la base de datos local.
Almacenamiento local	Store Registros Service	Store Messages	Insert Registro Mensaje	Este método obtiene los valores acumulados del conteo de mensajes de cada dos horas y almacena estos valores en la tabla de registro de mensajes.
		Store Calls	Insert Registro Llamada	Este método obtiene los valores acumulados del conteo de llamadas de cada dos horas y almacena estos valores en la tabla de registro de llamadas.
Envío de datos al servidor	Send Information Service	Store Information	Tarea Almacenar MSGSWS	En este método se recuperan los valores de la tabla local de registro de mensajes almacenados cada dos horas y se envían a la base de datos remota.

Módulo	Interfaz	Clase	Método	Descripción
			Tarea Almacenar CALLSWS	En este método se recuperan los valores de la tabla local de registro de llamadas que fueron almacenados cada dos horas y se envían a las base de datos remota.

4.5 Aplicación web

Esta aplicación permite que el usuario consulte las interacciones usuario-computadora y las interacciones de comunicación social que fueron monitoreadas. Las etapas que se incluyen en el desarrollo de la aplicación están basadas en el modelo en cascada. En la etapa de análisis se incluyen los requerimientos funcionales y no funcionales. En la etapa de diseño la arquitectura. Esta aplicación consulta la base de datos remota, por lo tanto, en este apartado no se incluye ningún modelo de base de datos particular para esta aplicación. En la etapa de implementación se describe la estructura de programación.

4.5.1 Análisis de la aplicación web

Los requerimientos funcionales para la aplicación web se describen en la Tabla 4.9. Respecto a los requerimientos relacionados con las tecnologías monitoreadas, los datos recuperados se descargan en formato PDF o formatos de imagen como JPEG, PNG y SVG. Debido a que toda la funcionalidad es de consulta esta aplicación interactúa directamente con la base de datos remota del sistema LaborCheck.

Tabla 4.9 Requerimientos funcionales de la aplicación web

Requerimiento funcional
La aplicación permite que el usuario inicie sesión para poder consultar los datos monitoreados.
La aplicación recupera los datos de las variables relacionadas con la tecnología teclado.
La aplicación recupera los datos de las variables relacionadas con la tecnología mouse.
La aplicación recupera los datos de las variables relacionadas con la tecnología ventanas.
La aplicación recupera los datos de las variables relacionadas con la tecnología mensajes.
La aplicación recupera los datos de las variables relacionadas con la tecnología llamadas.

Estos requerimientos son presentados gráficamente en el diagrama de casos de uso que se muestra en la Figura 4.11. La descripción general de los casos de uso es descrita en el capítulo 5 de pruebas y resultados. Por lo tanto, en este diagrama se muestra de manera general la funcionalidad de la aplicación web.

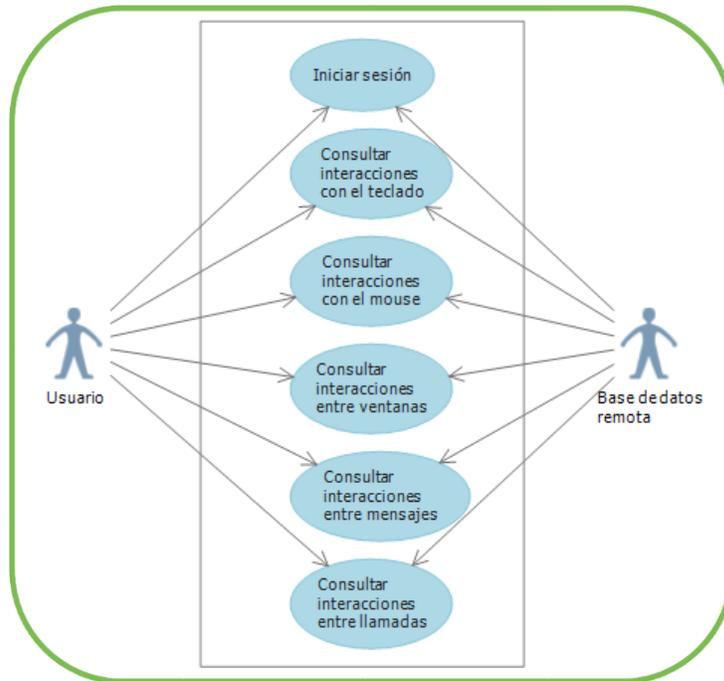


Figura 4.11 Diagrama de casos de uso de la aplicación web

Los requerimientos no funcionales se muestran en la Tabla 4.10. Están clasificados de acuerdo a dos categorías: requerimientos de software y confidencialidad.

Tabla 4.10 Requerimientos no funcionales de la aplicación web

Clasificación	Requerimiento
Software	La aplicación es consultada desde la aplicación de interacción usuario computadora.
	La aplicación es consultada desde la aplicación de interacción social.
	Los datos de la aplicación se muestran gráficamente.
	Las gráficas y el contenido de la aplicación se adapta al tamaño de la pantalla de cualquier dispositivo que cuente con un navegador web para consultar la aplicación.
Confidencialidad	Los datos de inicio de sesión pertenecen a un sólo usuario por lo tanto sólo él conoce su nombre de usuario y contraseña.

4.5.2 Diseño de la aplicación web

En esta sección se define el diseño de la arquitectura que se muestra en la Figura 4.12. En esta arquitectura se puede observar que todos los componentes que se incluyen en la aplicación web interactúan directamente con la base de datos remota. Por medio de esta base de datos se recuperan los valores de las variables de las tecnologías incluidas en esta investigación: teclado, mouse, ventanas, mensajes y llamadas.

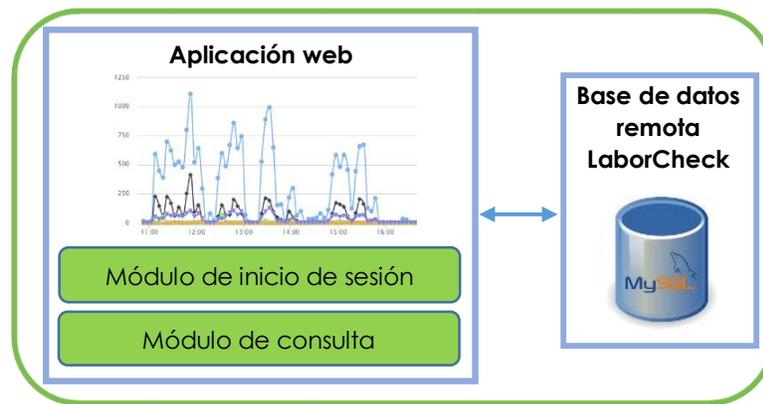


Figura 4.12 Arquitectura de la aplicación web

Módulo de inicio de sesión. En este módulo el usuario ingresa los datos de autenticación para que pueda consultar la información monitoreada de las variables usuario-computadora y de interacción social.

Módulo de consulta. En este módulo se muestra un campo de consulta que permite filtrar los datos de las tecnologías monitoreadas. El tipo de consulta es por día. Sin embargo, cada vez que el usuario entra a este módulo puede observar la información del día actual. La información es mostrada al usuario por medio de una gráfica de series por cada tecnología.

4.5.3 Implementación de la aplicación web

En esta sección se describe la implementación de la aplicación web, en la cual se mencionan los detalles técnicos y los modelos de programación. La pantalla principal de la aplicación web se puede observar en la Figura 4.13. Esta pantalla muestra la aplicación web consultada por medio de un *Smartphone* ajustando los componentes de la aplicación al tamaño de la pantalla.



Figura 4.13 Aplicación web

Esta aplicación está desarrollada sobre el entorno de programación *NetBeans* con el lenguaje de programación PHP. La aplicación utiliza el modelo-vista-controlador incorporado dentro del entorno de programación *Yii* en su versión 2.0. En esta aplicación se programaron diferentes métodos para recuperar los datos de las tecnologías, mostrados en la Tabla 4.11.

El módulo de inicio de sesión se implementa por medio de un formulario que utiliza la clase *usuario* para validar los datos introducidos en el formulario, se validan en caso de que aún no exista una sesión web activa con el mismo usuario.

El módulo de consulta está implementado por medio de un formulario web, que incorpora un campo de consulta de fechas sobre un componente *datepicker* de la librería *jQuery*. Los valores de las variables que se recuperan de cada tecnología se muestran al usuario por medio de gráficas de series que fueron programadas con la librería *Highcharts*.

Los valores que recibe la librería *Highcharts* están en formato JSON. Por lo tanto, los métodos programados envían esos valores con el mismo formato. En este formato se incluyen los lapsos de tiempo de monitoreo de las variables que son mostrados sobre el eje x y los valores sobre el eje y de las gráficas de series.

Tabla 4.11 Métodos programados para la aplicación web

Modulo	Formulario	Clase	Método	Descripción
Inicio de sesión	Login	Usuario	Validate Password	En este método se valida que la contraseña y el usuario existen en la base de datos remota.
Módulo de consulta	Chart	Series Chart	Obtener SerieTeclado	En este método se obtienen los valores de las variables del teclado y se añaden a una cadena con formato JSON que se envía al módulo de consulta.
			Obtener SerieMouse	En este método se obtienen los valores de las variables del mouse y se añaden a una cadena con formato JSON que es enviada al módulo de consulta.
			Obtener SerieVentanas	En este método se obtienen los valores de las variables de las ventanas y se añaden a una cadena con formato JSON que se envía al módulo de consulta.
			Obtener SerieMensajes	En este método se obtienen los valores de las variables de mensajes y se añaden a una cadena con formato JSON que se envía al módulo de consulta.
			Obtener SerieLlamadas	En este método se obtienen los valores de las variables de llamadas y se añaden a una cadena con formato JSON que se envía al módulo de consulta.

4.6 Interfaz de comunicación

La interfaz de comunicación permite que las aplicaciones de monitoreo se comuniquen por medio de servicios web. Estos servicios brindan a las aplicaciones servicios para insertar, actualizar y eliminar información dentro de la base de datos remota.

Los servicios web están programados sobre el Framework de programación *Yii2*. SOAP es el protocolo que permite la comunicación entre las aplicaciones y el servidor por medio de intercambio de datos XML. Estos servicios generan peticiones a la base de datos remota que utiliza el sistema gestor de base de datos MySQL.

Las peticiones hacia la base de datos remota se generan por medio de la interfaz *Active Record*, la cual inserta la información utilizando el modelo orientado a objetos. Para la inserción, borrado y actualización se utilizan transacciones, las cuales aseguran que la información que se envía a diferentes tablas se inserte sin errores.

Los servicios web que permiten realizar operaciones relacionadas con el usuario se muestran en la Tabla 4.12. Los servicios web que utiliza la aplicación de interacción usuario-computadora se muestran en la Tabla 4.13. Los servicios web que utiliza la aplicación de interacción social se muestran en la Tabla 4.14 y los servicios web que utiliza la aplicación web para mostrar la información de todas las tecnologías incluidas en el sistema LaborCheck se muestran en la Tabla 4.15.

Tabla 4.12 Servicios web para la gestión de usuarios

Entidad	Servicio	Descripción
Usuario	insertarUsuario(string)	Este servicio agrega un nuevo registro en la tabla usuarios. Los datos del usuario se reciben en una cadena con formato JSON, el valor de respuesta que se retorna es el <i>id</i> del usuario en caso de haber sido insertado correctamente o el valor de cero en caso de ocurrir un error al ser insertado. Lo utiliza la aplicación de interacción usuario-computadora en el formulario de registro.
	obtenerIdUsuario(string)	Este servicio obtiene el <i>id</i> del usuario enviando el nombre de usuario y contraseña. Estos datos se reciben en una cadena con formato JSON. Retorna el <i>id</i> recuperado o cero en caso de no existir en la base de datos. Lo utilizan las tres aplicaciones para autenticar al usuario.
	obtenerIdUsuario XUsuario(string)	Este servicio obtiene el <i>id</i> del usuario enviando el nombre de usuario. El valor que retorna es el <i>id</i> del usuario o cero en caso de no ser encontrado. Lo utiliza la aplicación de escritorio para validar el usuario.

Tabla 4.13 Servicios web para la aplicación de interacción usuario-computadora

Tecnología	Servicio	Descripción
Teclado	insertarRegistros Teclado(string)	Este servicio inserta las interacciones generadas con el teclado. El conjunto de datos se recibe en una cadena con formato JSON. El valor que retorna es de uno cuando los registros se insertan correctamente y cero en caso de haber ocurrido un error.
Mouse	insertarRegistros Mouse(string)	Este servicio inserta las interacciones generadas con el mouse. El conjunto de datos se recibe en una cadena con formato JSON. El valor que retorna es uno en caso de que los registros se inserten correctamente y en caso contrario retorna cero.
Ventanas	insertarRegistros Ventanas(string)	Este servicio inserta las interacciones generadas entre las ventanas de Windows. El conjunto de datos se recibe en una cadena con formato JSON. El valor que retorna es uno en caso de que se inserten correctamente y en caso contrario retorna cero.

Tabla 4.14 Servicios web para la aplicación de interacción social

Tecnología	Servicio	Descripción
Mensajes	insertarRegistros MSGS(string)	Este servicio inserta las interacciones de comunicación que se generan a través de mensajes. Los datos se reciben en una cadena JSON. El valor de retorno es uno en caso de ser insertados correctamente y en caso contrario devuelve el valor de cero.
Llamadas	insertarRegistros CALLS(string)	Este servicio inserta las interacciones de comunicación que se generan a través de llamadas. Los datos se reciben en una cadena JSON. El valor que retorna es uno en caso de ser insertados correctamente y en caso contrario devuelve cero.

Tabla 4.15 Servicios web para la aplicación web

Tecnología	Servicio	Descripción
Teclado	obtenerRegistros Teclado(string)	Este servicio recupera de la base de datos remota los valores de la tecnología teclado. Los datos para filtrar la consulta se reciben en una cadena JSON. De igual forma, los registros que se recuperan se retornan a través de una cadena con formato JSON.
Mouse	obtenerRegistros Mouse(string)	Este servicio recupera de la base de datos remota los valores de la tecnología mouse. Los datos para filtrar la consulta se obtienen de un JSON. Los valores recuperados del mouse se devuelven por medio de una cadena en formato JSON.
Ventanas	obtenerRegistros Ventanas(string)	Este servicio recupera de la base de datos remota los valores de la tecnología ventanas. Los datos que recibe para filtrar la consulta se presentan en una cadena JSON. De la misma forma, retorna los datos recuperados por medio de una cadena con formato JSON.
Mensajes	obtenerRegistros Mensajes(string)	Este servicio recupera de la base de datos remota los valores de la tecnología mensajes. Los datos para filtrar la consulta se reciben en una cadena con formato JSON. Igualmente, retorna los valores recuperados por medio de una cadena con formato JSON.
Llamadas	obtenerRegistros Llamadas(string)	Este servicio recupera de la base de datos remota los valores de la tecnología llamadas. Los datos para filtrar la consulta se reciben en una cadena con formato JSON. Asimismo, retorna los registros recuperados en una cadena con formato JSON.

4.7 Base de datos remota

El modelo relacional de la base de datos remota se muestra en la Figura 4.14. Con base en este modelo se almacena la información de las tecnologías monitoreadas: teclado, mouse, ventanas, mensajes y llamadas. Esta base de datos reside sobre un servidor web en la nube que tiene instalado el sistema gestor de base de datos MySQL.

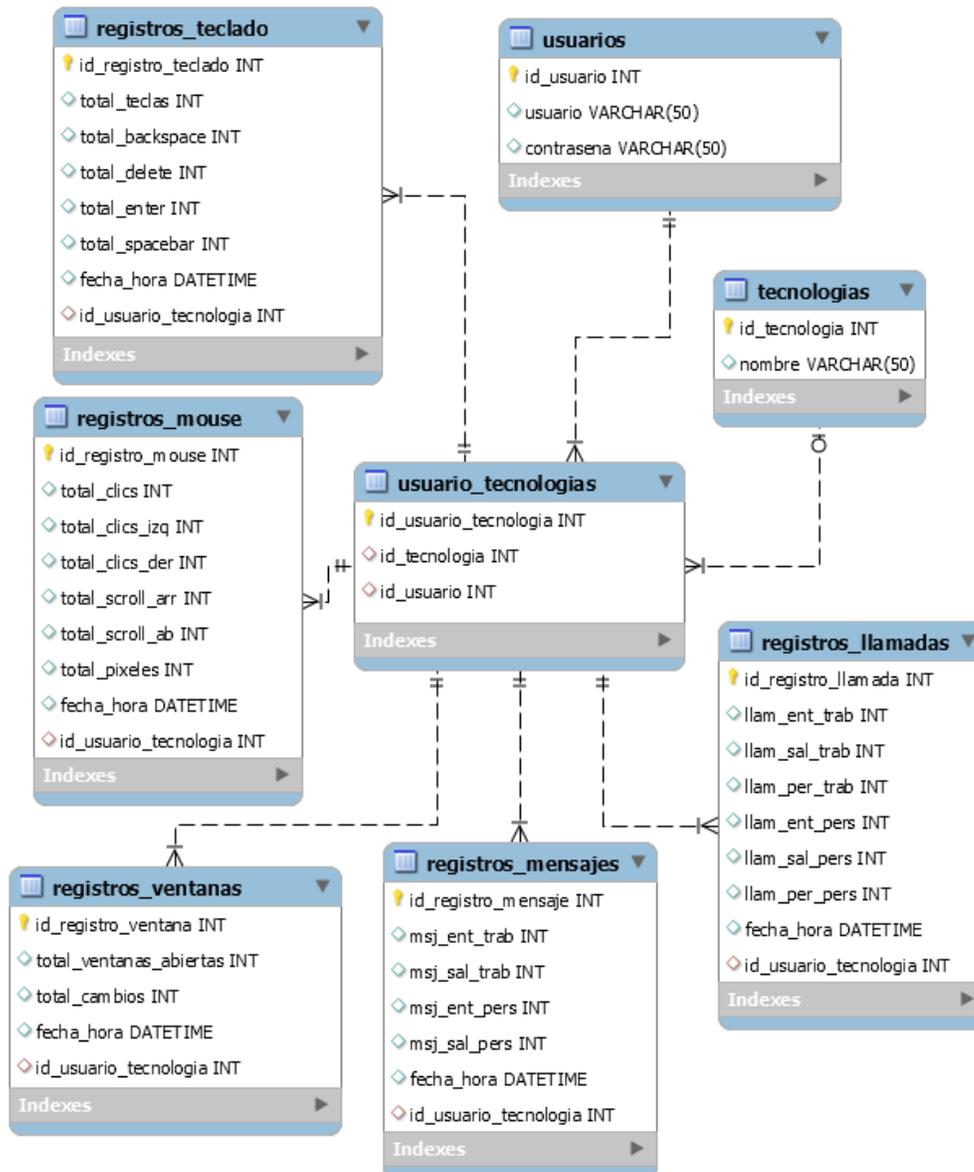


Figura 4.14 Diagrama entidad relación para la base de datos remota del sistema LaborCheck

En esta base de datos se reciben los valores de las variables que son registrados localmente cada cinco minutos en la aplicación de interacción usuario-computadora. De igual manera, se reciben los valores de las variables que son registrados localmente cada dos horas en la aplicación de interacción social. En el anexo C se describen las tablas del diagrama ER.

Capítulo 5

Pruebas y Resultados

Capítulo 5. Pruebas y Resultados

En este capítulo se integran los planes de prueba de las aplicaciones que son parte del sistema LaborCheck: aplicación de interacción usuario-computadora, aplicación de interacción social y aplicación web. En cada plan de pruebas se comprueba la funcionalidad de cada aplicación. Estos planes de prueba incluyen algunas especificaciones del estándar IEEE 829-2008, el cual fue diseñado para documentar las pruebas funcionales que se realizan a un sistema de software. Además, se compara la precisión del sistema LaborCheck con otro software que también recupera las interacciones incluidas en esta investigación.

5.1 Plan de pruebas para la aplicación de interacción usuario-computadora

Las pruebas planteadas en este documento muestran los resultados de la funcionalidad de los módulos desarrollados en la aplicación de interacción usuario-computadora y comprueban su comportamiento dentro del entorno de implementación.

5.1.1 Elementos de la prueba

Antes de llevar a cabo las pruebas es necesario cumplir con un conjunto de elementos:

- Tener un teclado para monitorear las interacciones que se producen con este dispositivo.
- Tener un mouse que incluya la rueda de desplazamiento, debido a que algunas de las variables que se monitorean son la cantidad de desplazamientos producidos con este elemento del mouse.

Los casos de prueba deberán ajustarse al formato establecido:

Caso de prueba: Número del caso
Resultado: Aprobado/No aprobado
Desarrollo de la prueba. Describe el procedimiento detallado para generar la prueba así como la salida esperada.
Evidencias del desarrollo de la prueba. Muestra las evidencias por medio de capturas de pantalla que demuestran la funcionalidad de la prueba .
Observaciones. Describen los sucesos o inconvenientes que se observen al ejecutar el caso de prueba.

5.1.2 Características probadas

Las características de funcionalidad evaluadas con la aplicación se describen en este apartado, las cuales corresponden con los casos de uso generados en la etapa de análisis:

1. **Registro de usuario.** Consiste en ingresar los datos requeridos para registrar un nuevo usuario, validar que los campos no se dejen vacíos, y minimizar la aplicación en la barra de tareas.
2. **Inicio de sesión.** Consiste en autenticar la cuenta de usuario mediante el nombre de usuario y una contraseña válida, validar que los campos no se ingresen vacíos y que la aplicación se minimice en la barra de tareas.
3. **Reinicio automático de la aplicación.** Consiste en comprobar que la aplicación se ejecute automáticamente después de reiniciar el sistema operativo Windows.
4. **Interacciones con el teclado.** Consiste en contabilizar las teclas presionadas, teclas de tipo retroceso, teclas de tipo eliminar, teclas de tipo *enter* y teclas presionadas con la barra espaciadora.
5. **Interacciones con el mouse.** Consiste en contabilizar los movimientos generados con el mouse: total de clics, clics izquierdos, clics derechos, desplazamientos hacia arriba con la rueda del mouse, desplazamientos hacia abajo con la rueda del mouse y total de píxeles recorridos con el puntero del mouse.
6. **Interacciones con ventanas.** Consiste en contabilizar las ventanas abiertas y los cambios entre ventanas.
7. **Almacenamiento local.** Consiste en almacenar la cantidad de interacciones generadas con el teclado, el mouse y ventanas. Este almacenamiento se genera cada 5 minutos en la base de datos local.
8. **Almacenamiento remoto.** Consiste en obtener los registros de la base de datos local y enviarlos a la base de datos remota. Esta operación se realiza cada 17 minutos, por lo tanto, se enviarán hacia la base de datos remota un total de 3 registros de cada una de las tecnologías: teclado, mouse y ventanas.

Si alguna de las características anteriores no se cumple, se marcará como "no aprobada" y se revisarán los aspectos técnicos para corregir los posibles errores.

5.1.3 Enfoque

En estas pruebas se verifica que los datos de entrada de los formularios estén válidos correctamente y que la aplicación sea capaz de recolectar la información de los dispositivos de entrada como: teclado y mouse, enviando correctamente esta información hacia la base de datos remota, comprobando la funcionalidad de la aplicación a nivel externo que comúnmente se le denomina pruebas de caja negra.

5.1.4 Criterios de suspensión y requerimientos de reanudación

Las pruebas pueden ser suspendidas bajo ciertos criterios:

- En caso de que la computadora se apague o si es computadora portátil se quede sin batería o que entre en modo de suspensión.
- Cuando el sistema operativo Windows se vuelva inestable y no responda a las interacciones del usuario.

Para la reanudación de las pruebas es necesario que la interrupción que detuvo el proceso sea resuelta.

5.1.5 Requisitos ambientales

En esta sección se mencionan los requisitos necesarios de hardware y software para que las pruebas se realicen correctamente:

Hardware:

- Teclado de cualquier marca.
- Mouse con rueda de desplazamiento de cualquier marca.

Software:

- Sistema operativo Windows versión 7, versión 8 o versión 10.
- Microsoft .NET Framework 4.5

5.1.6 Responsabilidades

El Ingeniero en Informática Joan Misael Camaños Anacleto es el encargado de realizar cada una de las pruebas mencionadas en el documento. Además de elaborar la documentación y corrección de los errores detectados.

El presente plan de pruebas es aprobado por la Dra. Alicia Martínez Rebollar investigadora en la especialidad de Sistemas Distribuidos del Centro Nacional de Investigación y Desarrollo Tecnológico.

5.1.7 Nomenclatura

Cada prueba debe tener un identificador único para hacer referencia a las características probadas. Se utilizan las iniciales CP para referirse al caso de prueba, AUC a la aplicación de interacción usuario-computadora más el número consecutivo de las pruebas.

Nombre del caso de prueba	Nomenclatura
Registro de usuario	CP_AUC_1
Inicio de sesión	CP_AUC_2
Reinicio automático de la aplicación	CP_AUC_3
Interacciones con el teclado	CP_AUC_4
Interacciones con el mouse	CP_AUC_5
Interacciones con ventanas	CP_AUC_6
Almacenamiento local	CP_AUC_7
Almacenamiento remoto	CP_AUC_8

5.1.8 Casos de prueba

Los casos de prueba CP_AUC_1 al CP_AUC_3, CP_AUC_7 y CP_AUC_8 están incluidos en el anexo D. Estos casos muestran el proceso relacionado con usuarios, el reinicio automático de la

aplicación y el almacenamiento de la información. Por lo tanto, en esta sección se muestran los casos de prueba relacionados con la detección de las interacciones usuario-computadora en los casos de prueba CP_AUC_4 al CP_AUC_6.

CP_AUC_4

Aprobado

Desarrollo de la prueba:

1. Abrir diferentes programas y teclear varias palabras para contabilizar las interacciones con el teclado. En este caso se comprueba con el Bloc de Notas y un documento de Word. La pantalla de depuración de la aplicación deberá contabilizar cada uno de las teclas que se estén presionando en los diferentes programas.

Evidencias del desarrollo de la prueba:

1.

Observaciones: La cantidad de teclas presionadas sobre el teclado corresponde con la cantidad obtenida por la aplicación de interacción usuario-computadora.

CP_AUC_5

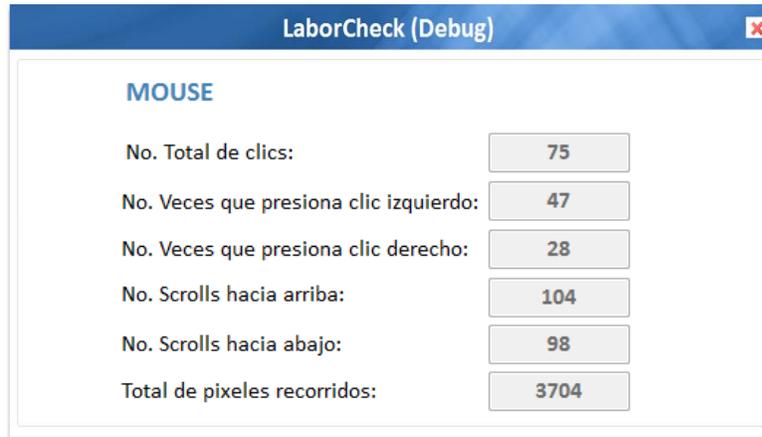
Aprobado

Desarrollo de la prueba:

1. El usuario deberá interactuar con el mouse; las interacciones que se deben llevar a cabo son: presionar el botón derecho e izquierdo del mouse, girar la rueda de desplazamiento en forma vertical hacia arriba y hacia abajo, mover el puntero del mouse dentro de la pantalla de la computadora. El sistema deberá contabilizar cada uno de los movimientos generados con el dispositivo y mostrar los resultados en el formulario de depuración.

Evidencias del desarrollo de la prueba:

1.



Observaciones: Los movimientos generados con el mouse se contabilizaron correctamente por la aplicación de monitoreo de interacciones usuario-computadora.

CP_AUC_6

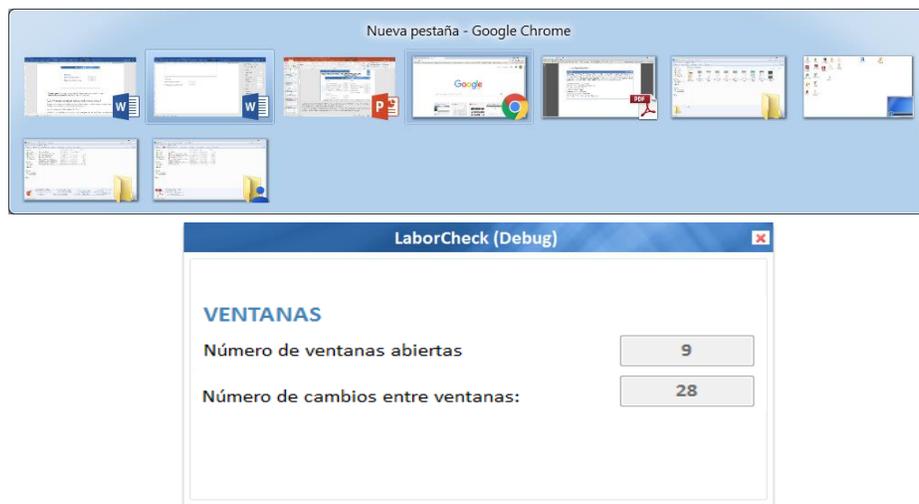
Aprobado

Desarrollo de la prueba:

1. Abrir diferentes ventanas dentro del área de trabajo de la pantalla, ajustar el tamaño de las ventanas de tal manera que se puedan visualizar todas y hacer cambios entre ellas con el puntero del mouse, también hacer cambios entre ventanas usando la combinación de teclas *Alt + Tab*. La aplicación deberá contabilizar cada una de las ventanas abiertas y los cambios entre ellas, mostrando los resultados en la pantalla de depuración.

Evidencias del desarrollo de la prueba:

1.



Observaciones: La aplicación detectó correctamente la creación de nuevas ventanas y los cambios generados con el mouse y con la combinación de teclas *Alt+Tab*.

5.2 Plan de pruebas para la aplicación de interacción social

Las pruebas planteadas en este documento muestran los resultados de la funcionalidad de los módulos desarrollados en la aplicación móvil, se comprueba su comportamiento dentro del entorno de implementación.

5.2.1 Elementos de la prueba

Antes de llevar a cabo las pruebas es necesario cumplir con un conjunto de elementos:

- Tener un *Smartphone* con sistema operativo Android.
- Instalar la aplicación en el dispositivo móvil.

Los casos de prueba deberán ajustarse al formato establecido:

Caso de prueba: Número de caso
Resultado: Aprobado/No aprobado
Desarrollo de la prueba. Describe el procedimiento detallado para generar la prueba y se menciona la salida esperada.
Evidencias del desarrollo de la prueba. Muestra las evidencias por medio de capturas de pantalla que demuestran la funcionalidad de la prueba.
Observaciones. Describe los sucesos o inconvenientes que se observen al ejecutar el caso de prueba.

5.2.2 Características probadas

Las características de funcionalidad evaluadas con la aplicación se describen en esta sección:

1. **Inicio de sesión.** Consiste en autenticar al usuario mediante un nombre de usuario y una contraseña, validar que los campos no se ingresen vacíos y validar que el usuario y contraseña existan en la base de datos remota.
2. **Reinicio automático de la aplicación.** Consiste en comprobar que los servicios de detección de mensajes, almacenamiento local y almacenamiento remoto se ejecuten automáticamente después de reiniciar el sistema operativo Android.
3. **Registrar contacto.** Consiste en capturar los datos de un nuevo contacto para agregarse a la lista de contactos del usuario que está siendo monitoreado. Sólo se monitorea la comunicación con los contactos registrados en este apartado.
4. **Editar contacto.** Consiste en actualizar la información del contacto agregado en la lista de contactos del usuario. Los cambios se reflejan en las tablas de la base de datos local.
5. **Eliminar contacto.** Consiste en borrar el contacto de la base de datos local, mostrando un cuadro de dialogo al usuario en donde se confirma la eliminación del contacto de forma permanente.
6. **Detección de mensajes.** Consiste en detectar los cambios generados en la base de datos del registro de mensajes del dispositivo móvil. Los cambios detectados serán mensajes recibidos y enviados. Cada uno de estos cambios detectados serán almacenados en la base de datos local.

7. **Detección de llamadas.** Consiste en detectar los cambios generados en la base de datos del registro de llamadas del dispositivo móvil. Los cambios detectados serán llamadas realizadas, perdidas y recibidas. Cada uno de estos cambios detectados serán almacenados en la base de datos local.
8. **Almacenamiento local.** Consiste en almacenar la cantidad de cambios detectados de mensajes y llamadas cada dos horas en la base de datos local.
9. **Almacenamiento remoto.** Consiste en enviar los registros almacenados en la base de datos local hacia la base de datos remota. Esta operación se realiza cada dos horas y media.

Si alguna de las características anteriores no se cumple, se marcará como "no aprobada" y se revisaran los aspectos técnicos para corregir los posibles errores.

5.2.3 Enfoque

En estas pruebas se verifica que los datos de entrada capturados en la aplicación móvil se validen correctamente y que la aplicación sea capaz de recolectar la información de mensajes y llamadas, enviando correctamente esta información hacia la base de datos remota, comprobando así la funcionalidad de la aplicación a nivel externo que se le denomina prueba de caja negra.

5.2.4 Criterios de suspensión y requerimientos de reanudación

Las pruebas pueden ser suspendidas bajo ciertos criterios:

- En caso de que el dispositivo móvil se apague manualmente o por falta de batería.
- Cuando el sistema operativo Android se vuelva inestable y no responda a las peticiones del usuario.

Para la reanudación de las pruebas es necesario que la interrupción que detuvo el proceso de pruebas sea resuelta.

5.2.5 Requisitos ambientales

En esta sección se mencionan los requisitos necesarios de hardware y software para que las pruebas se realicen apropiadamente:

Hardware:

- *Smartphone.*

Software:

- Sistema operativo Android.

5.2.6 Responsabilidades

El Ingeniero en Informática Joan Misael Camaños Anacleto es el encargado de realizar cada una de las pruebas mencionadas en el documento. Además de elaborar la documentación y corrección de los errores detectados.

El presente plan de pruebas es aprobado por la Dra. Alicia Martínez Rebollar investigadora en la especialidad de Sistemas Distribuidos del Centro Nacional de Investigación y Desarrollo Tecnológico.

5.2.7 Nomenclatura

Cada prueba debe tener un identificador único para hacer referencia a las características probadas. Se utilizan las iniciales CP para referenciarse al caso de prueba, AS a la aplicación de interacción social más el numero consecutivo de las pruebas.

Nombre del caso de prueba	Nomenclatura
Inicio de sesión	CP_AS_1
Reinicio automático de la aplicación	CP_AS_2
Registrar contacto	CP_AS_3
Editar contacto	CP_AS_4
Eliminar contacto	CP_AS_5
Detección de mensajes	CP_AS_6
Detección de llamadas	CP_AS_7
Almacenamiento local	CP_AS_8
Almacenamiento remoto	CP_AS_9

5.2.8 Casos de prueba

Los casos de prueba CP_AS_1 al CP_AS_5, CP_AS_8 y CP_AS_9 están incluidos en el anexo E. Estos casos muestran el proceso relacionado con inicio de sesión, reinicio automático de la aplicación, administración de contactos y almacenamiento de la información. Por lo tanto, en esta sección se muestran los casos de prueba para la detección de las interacciones de comunicación entre el usuario y sus contactos en los casos de prueba CP_AS_6 y CP_AS_7.

CP_AS_6

Aprobado

Desarrollo de la prueba:

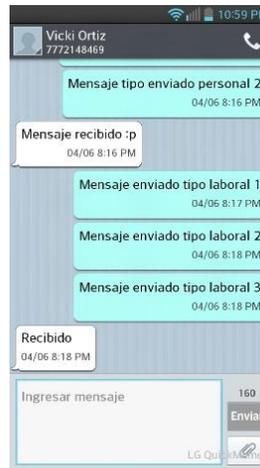
1. Comprobar que el servicio que detecta los mensajes se encuentre en ejecución, el cual se puede comprobar al abrir el menú de ajustes del dispositivo móvil, dar clic en el administrador de aplicaciones y seleccionar la pestaña de "Aplicaciones en ejecución". El sistema Android mostrará las aplicaciones que se están ejecutando en el sistema operativo, seleccionar la aplicación "LaborCheck" y comprobar que el servicio nombrado "PhoneService" se encuentra en ejecución.
2. Comprobar la funcionalidad con los mensajes. Enviar mensajes de prueba y solicitar al mismo contacto contestar el mensaje. Los mensajes se contabilizarán de acuerdo al tipo de contacto registrado ya sea como mensajes laborales o personales y clasificados de acuerdo a su categoría: recibidos o enviados.
3. Comprobar que los registros de conteo de mensajes se actualicen correctamente. Utilizar alguna aplicación para gestionar la base de datos local en el dispositivo móvil como por ejemplo "aSQLiteManager", el dispositivo móvil deberá estar configurado en modo "root" debido a que la información de las aplicaciones instaladas se encuentra protegida por el sistema operativo Android por motivos de seguridad. Una vez configurado el dispositivo, comprobar que las pruebas en mensajes se encuentran debidamente almacenadas en la tabla "conteomensajes".

Evidencias del desarrollo de la prueba:

1.



2.



3.

Tabla "conteomensajes"

id	id_usuario	msgs_rec_lab	msgs_rec_per	msgs_env_lab	msgs_env_per	id_ultimo_mensaje
1	15	1	1	3	2	394

Observaciones: Los mensajes se detectaron correctamente de acuerdo al tipo de contacto y a la categoría correspondiente.

CP_AS_7

Aprobado

Desarrollo de la prueba:

1. Comprobar que el servicio que detecta las llamadas se encuentra en ejecución, el cual se puede observar al dar clic en el administrador de aplicaciones y seleccionar la pestaña de "Aplicaciones en ejecución". El sistema Android mostrará las aplicaciones que se están ejecutando en el sistema operativo, seleccionar la aplicación "LaborCheck" y comprobar que el servicio nombrado "PhoneService" se encuentra en ejecución.
2. Comprobar la funcionalidad en llamadas. Generar llamadas con alguno de los contactos agregados en la lista de contactos. Las llamadas se contabilizarán de acuerdo al tipo de contacto registrado ya sea como personales o laborales y clasificadas de acuerdo a su categoría: realizadas, recibidas y perdidas.
3. Comprobar que los registros se actualicen correctamente. Utilizar alguna aplicación para gestionar la base de datos local en el dispositivo móvil como por ejemplo "aSQLiteManager", el dispositivo móvil deberá estar configurado en modo "root" debido a que la información de las aplicaciones instaladas se encuentra protegida por el sistema operativo Android por motivos de seguridad. Una vez configurado el dispositivo, comprobar que las pruebas con llamadas se encuentran debidamente almacenadas en la tabla "conteollamadas".

Evidencias del desarrollo de la prueba:

1.



2.



3.

Tabla "conteollamadas"

id	id_usuario	calls_sal_lab	calls_sal_per	calls_ent_lab	calls_ent_per	calls_perd_lab	calls_perd_per	id_ultima_llamada
1	15	3	2	1	1	1	1	1485

Observaciones: Las llamadas se detectaron correctamente de acuerdo al tipo de contacto y la categoría correspondiente.

5.3 Plan de pruebas para la aplicación web

Las pruebas planteadas en este documento muestran los resultados de la funcionalidad de los módulos desarrollados en la aplicación web, se comprueba su comportamiento dentro del entorno de implementación.

5.3.1 Elementos de la prueba

Antes de llevar a cabo las pruebas es necesario cumplir con un conjunto de elementos:

- Tener una computadora o dispositivo móvil.
- Tener instalado un navegador web.
- Ingresar la dirección web de la aplicación en la barra de direcciones del navegador.

Los casos de prueba deberán ajustarse al formato establecido:

Caso de prueba: Número de caso
Resultado: Aprobado/No aprobado
Desarrollo de la prueba: Describe el procedimiento detallado para generar la prueba y se menciona la salida esperada.
Evidencias del desarrollo de la prueba: Muestra las evidencias por medio de capturas de pantalla que demuestran la funcionalidad de la prueba.
Observaciones: Describen los sucesos o inconvenientes que se observen al ejecutar el caso de prueba.

5.3.2 Características probadas

Las características de funcionalidad evaluadas con la aplicación se describen en este apartado:

1. **Inicio de sesión.** Consiste en autenticar al usuario en la aplicación web mediante un nombre de usuario y una contraseña.
2. **Variables del teclado.** Consiste en recuperar la información consultada por día de las variables del teclado recolectada por medio de la aplicación de interacción usuario-computadora y mostrarla en una gráfica de series.
3. **Variables del mouse.** Consiste en recuperar la información consultada por día de las variables del mouse recolectada por medio de la aplicación de interacción usuario-computadora y mostrarla en una gráfica de series.
4. **Variables de ventanas.** Consiste en recuperar la información consultada por día de las variables de interacciones entre ventanas recolectada por medio de la aplicación de interacción usuario-computadora y mostrarla en una gráfica de series.
5. **Variables de mensajes.** Consiste en recuperar la información consultada por día de las variables de mensajes recolectada por medio de la aplicación de interacción social y mostrarla en una gráfica de series.

6. **Variables de llamadas.** Consiste en recuperar la información consultada por día de las variables de llamadas recolectada por medio de la aplicación de interacción social y mostrarla en una gráfica de series.

Si alguna de las características anteriores no se cumple, se marcará como “no aprobada” y se revisaran los aspectos técnicos para corregir los posibles errores.

5.3.3 Enfoque

En estas pruebas se verifica que la aplicación recupere la información recolectada de la aplicación de interacción usuario-computadora y de la aplicación de interacción social, así como desplegar correctamente la aplicación web dentro de una computadora o un *Smartphone*. Se comprueba la funcionalidad de la aplicación a nivel externo que se le denomina pruebas de caja negra.

5.3.4 Criterios de suspensión y requerimientos de reanudación

Las pruebas pueden ser suspendidas bajo los siguientes criterios:

- Cuando el navegador web se vuelva inestable y no responda a las peticiones del usuario.
- Cuando se suspenda la conexión a internet.

Para la reanudación de las pruebas es necesario que la interrupción que detuvo el proceso de pruebas sea resuelta.

5.3.5 Requisitos ambientales

En esta sección se mencionan los requisitos necesarios de hardware y software para que las pruebas se realicen apropiadamente:

Hardware:

- *Smartphone* o computadora.

Software:

- Navegador web.

5.3.6 Responsabilidades

El Ingeniero en Informática Joan Misael Camaños Anacleto es el encargado de realizar cada uno de las pruebas mencionadas en el documento. Además de elaborar la documentación y corrección de los errores detectados.

El plan de pruebas es aprobado por la Dra. Alicia Martínez Rebollar investigadora en la especialidad de Sistemas Distribuidos del Centro Nacional de Investigación y Desarrollo Tecnológico.

5.3.7 Nomenclatura

Cada prueba debe tener un identificador único para hacer referencia a las características probadas. Se utilizan las iniciales CP para referenciarse al caso de prueba, AW a la aplicación web más el número consecutivo de la prueba.

Nombre del caso de prueba	Nomenclatura
Inicio de sesión	CP_AW_1
Variables del teclado	CP_AW_2
Variables del mouse	CP_AW_3
Variables de ventanas	CP_AW_4
Variables de mensajes	CP_AW_5
Variables del llamadas	CP_AW_6

5.3.8 Casos de prueba

El caso de prueba CP_AW_1 está incluido en el anexo F. Este caso muestra el proceso relacionado con el inicio de sesión en la aplicación. De manera que, las gráficas que muestran la información almacenada se describen en esta sección con los casos de prueba CP_AW_2 al CP_AW_6.

CP_AW_2

Aprobado

Desarrollo de la prueba:

1. Dar clic sobre el campo fecha para seleccionar alguna fecha de consulta. La aplicación mostrará un calendario para que el usuario seleccione la fecha a consultar (esta opción estará disponible en el resto de las gráficas de la aplicación web).
2. Dar clic sobre el botón "Consultar". La aplicación recuperará los registros que fueron capturados cada cinco minutos por la aplicación de interacción usuario-computadora durante la fecha consultada. La información será mostrada en una gráfica de series que contiene las variables monitoreadas del teclado: total tecleos, tecla retroceso, tecla eliminar, tecla enter y tecla espaciadora. El usuario podrá navegar por cada uno de los puntos de la gráfica, los cuales corresponden a un registro almacenado y comprobar la cantidad de interacciones recuperadas de la variable sobre la que posicione el mouse, así como la hora exacta de registro (esta opción estará disponible en el resto de las gráficas de la aplicación web).
3. Dar clic sobre alguna de las variables mostradas en la parte inferior de la gráfica. La aplicación deberá habilitar o deshabilitar la información relacionada con la variable o las variables a las que se le dé clic, con lo cual se podrá apreciar mejor la información de la variable o variables de mayor interés para el usuario (esta funcionalidad estará disponible en el resto de las gráficas de la aplicación web).
4. Dar clic en el menú ubicado en la parte superior derecha de la gráfica. La aplicación mostrará un menú de opciones para poder imprimir la gráfica o descargarla en formato PNG, JPEG, PDF o SVG (esta opción estará disponible en el resto de las gráficas de la aplicación web).

Evidencias del desarrollo de la prueba:

1.



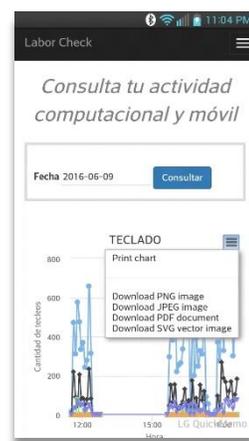
2.



3.



4.



Observaciones: Los valores de las variables recuperadas de la tecnología teclado se mostraron correctamente en la gráfica de series.

CP_AW_3

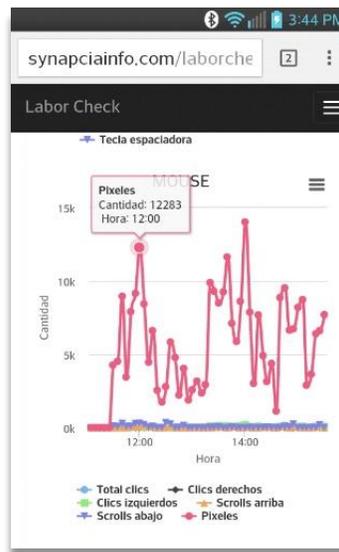
Aprobado

Desarrollo de la prueba:

1. Dar clic sobre el botón “Consultar” para recuperar los registros que fueron capturados cada cinco minutos por la aplicación de interacción usuario-computadora durante la fecha consultada. La información será mostrada en una gráfica de series que contiene las variables monitoreadas del mouse: total clics, clics derechos, clics izquierdos, scrolls hacia arriba, scrolls hacia abajo y cantidad de pixeles.

Evidencias del desarrollo de la prueba:

1.



Observaciones: La aplicación recuperó correctamente los valores de la tecnología Mouse y los mostró en la gráfica de series.

CP_AW_4

Aprobado

Desarrollo de la prueba:

1. Dar clic sobre el botón "Consultar" para recuperar los registros que fueron capturados cada cinco minutos por la aplicación de interacción usuario-computadora durante la fecha consultada. La información será mostrada en una gráfica de series que contiene las variables monitoreadas de las interacciones con ventanas: ventanas abiertas y cambios entre ventanas.

Evidencias del desarrollo de la prueba:

1.



Observaciones: La aplicación recuperó correctamente los valores de las variables relacionadas con la tecnología ventanas y las mostró en una gráfica de series dinámica.

CP_AW_5

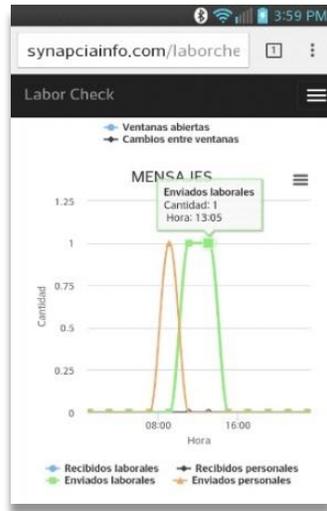
Aprobado

Desarrollo de la prueba:

1. Dar clic sobre el botón "Consultar" para recuperar los registros que fueron capturados cada dos horas por la aplicación de interacción social durante la fecha consultada. La información será mostrada en una gráfica de series que contiene las variables monitoreadas de mensajes: recibidos laborales, recibidos personales, enviados laborales y enviados personales.

Evidencias del desarrollo de la prueba:

1.



Observaciones: La aplicación recuperó correctamente los valores de las variables de comunicación entre el usuario y su contactos que se realiza por medio de mensajes.

CP_AW_6

Aprobado

Desarrollo de la prueba:

1. Dar clic sobre el botón "Consultar". La aplicación recuperara los registros que fueron capturados cada 2 horas por la aplicación móvil durante la fecha consultada. La información será mostrada en una gráfica de series que contiene las variables monitoreadas de llamadas: recibidas laborales, recibidas personales, realizadas laborales, realizadas personales, perdidas laborales y perdidas personales.

Evidencias del desarrollo de la prueba:

1.



Observaciones: La aplicación recuperó correctamente los valores de las variables de comunicación entre el usuario y sus contactos que se realiza por medio de llamadas.

5.4 Pruebas con software relacionado

En esta sección se muestran las pruebas que se realizaron con otras aplicaciones que obtienen información relacionada con las tecnologías de esta investigación. La precisión del sistema LaborCheck se comparó con tres aplicaciones que recuperan eventos en común: *Actual Keylogger*, *Mouse Recorder* y *HoverWatch*. La primera aplicación recupera los eventos relacionados con la tecnología teclado [33]. La segunda aplicación los eventos relacionados con la tecnología mouse y ventanas [34]. La última aplicación detecta mensajes y llamadas desde un *Smartphone* con sistema operativo Android [35].

Las interacciones generadas por medio del teclado se compararon con el software que se muestra en la Figura 5.1. Es una aplicación que almacena el nombre de las teclas presionadas obtenidas dentro del programa en el que se está trabajando. El reporte de las interacciones recuperadas en este software se generó por medio de un archivo de texto. Sobre este archivo se filtraron los distintos tipos de teclas para contabilizarlos. Después de haber filtrado la información, la cantidad de teclas de tipo retroceso, eliminar, *enter* y con la barra espaciadora obtenidas con el software relacionado recuperó las mismas cantidades obtenidas por la aplicación de interacción usuario-computadora.

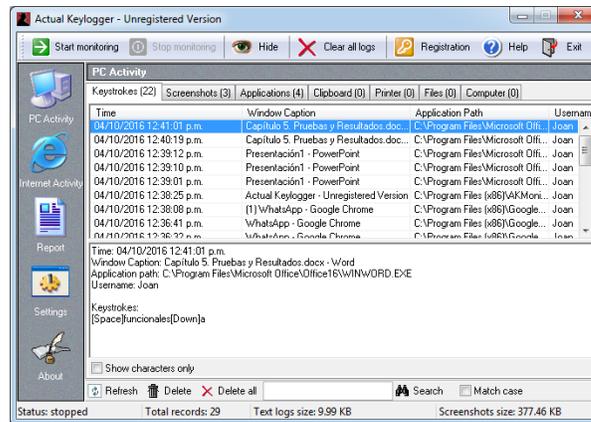


Figura 5.1 Aplicación relacionada con la tecnología teclado

Las interacciones generadas por medio del mouse y cambios entre ventanas se compararon con los resultados obtenidos por el software que se muestra en la Figura 5.2. En este software se genera un reporte incluido dentro de la misma aplicación, en donde se clasificaron los tipos de eventos que se recuperan desde el mouse. Estos eventos fueron contabilizados y comparados con los eventos recuperados por el sistema LaborCheck. Se comprobó que los eventos generados por el mouse y los eventos generados por la aplicación usuario-computadora coinciden en cantidad.

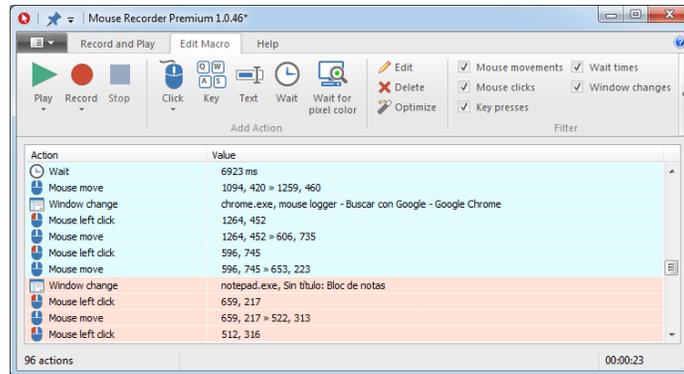


Figura 5.2 Aplicación relacionada con la tecnología mouse y ventanas

Las interacciones de comunicación social generadas por medio de los mensajes y llamadas se compararon con el software que se muestra en la Figura 5.3. Este software recupera los tipos de mensajes y llamadas generados en un dispositivo móvil con sistema operativo Android. Estos mensajes y llamadas se contabilizaron manualmente y se compararon con los que se contabilizan automáticamente con la aplicación de interacción social en donde el conteo manual y el conteo automático coinciden en cantidad por ambas aplicaciones.

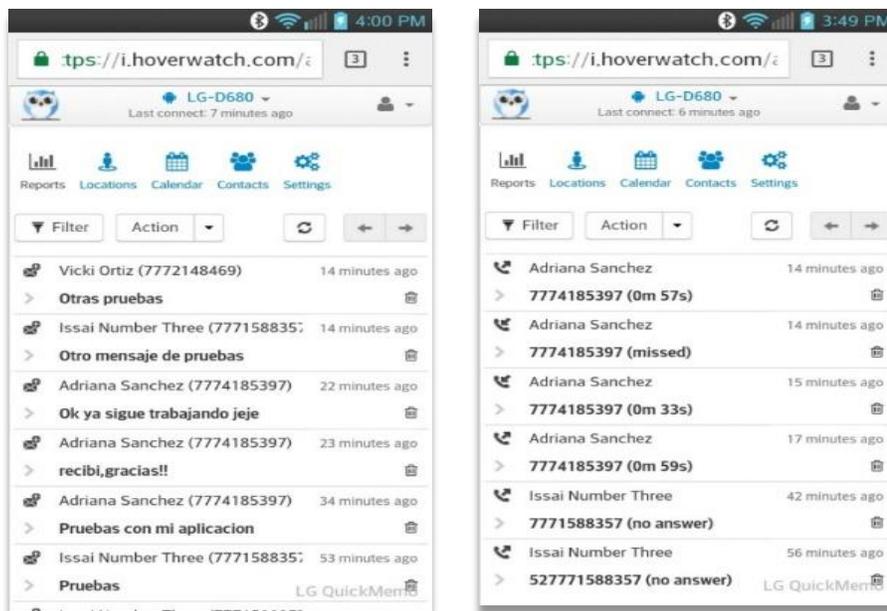


Figura 5.3 Aplicación relacionada con la tecnología de mensajes y llamadas

5.5 Resultados

Las pruebas se realizaron con seis usuarios durante tres meses. Las aplicaciones desarrolladas fueron instaladas en los dispositivos de los usuarios para monitorearlos simultáneamente. Las aplicaciones del sistema LaborCheck se ejecutaron sin presentar problemas para el usuario mientras se mantenía realizando sus actividades diarias. La información de cada usuario fue almacenada correctamente en la base de datos remota.

La aplicación de interacción usuario-computadora se instaló en las siguientes versiones de Windows: Windows 10, Windows 8 y Windows 7. Los antivirus instalados en estos equipos fueron: AVAST y AVG, los cuales no denegaron ningún acceso para poder llevar a cabo la recolección de la información de las variables del teclado, mouse y ventanas.

Las variables que presentaron mayor actividad durante los meses de evaluación con la tecnología teclado se muestra en la Figura 5.4. En la cual se puede observar que la tecla retroceso se presionó con mayor frecuencia y la tecla suprimir presentó poca actividad.

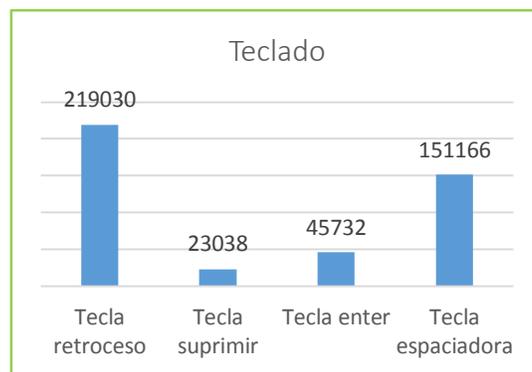


Figura 5.4 Actividad de las variables de la tecnología teclado

La actividad de las variables relacionadas con la tecnología mouse se puede apreciar en la Figura 5.5. La variable que presentó mayor actividad fue la cantidad de pixeles seguido del uso de la rueda del mouse y la que presentó menor actividad fue el uso del clic derecho.

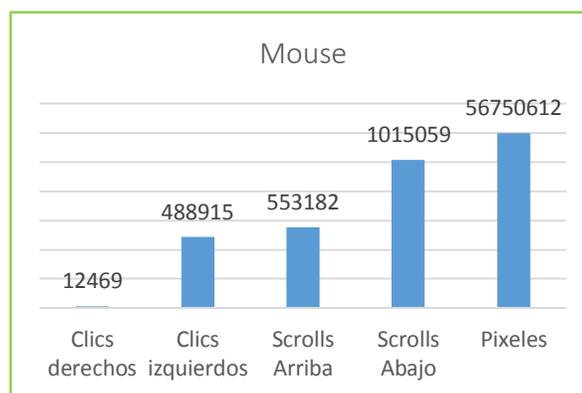


Figura 5.5 Actividad de las variables de la tecnología mouse

La actividad de las variables relacionadas con la interacción entre ventanas de Windows se muestra en la Figura 5.6. La variable que presentó mayor actividad fue la cantidad de cambios entre ventanas.

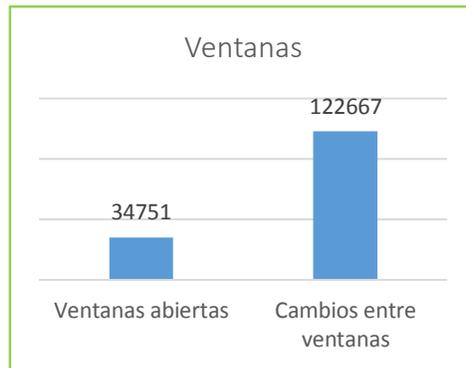


Figura 5.6 Actividad de las variables de la tecnología ventanas

La aplicación de interacción social fue instalada en las versiones de Android: 4.1.2., 4.3., 5.02 y 6.0.1, versiones que cubren aproximadamente un 90% del mercado de los dispositivos móviles con sistema operativo Android, este porcentaje se puede observar en el entorno de desarrollo Android Studio.

La actividad de las variables de mensajes y llamadas presentaron menor actividad en comparación con las variables de interacción usuario-computadora. La actividad de mensajes se puede apreciar en la Figura 5.7, en la cual se muestra una mayor comunicación personal con el uso de mensajes de texto. La actividad de llamadas en la Figura 5.8, la cual también muestra una actividad personal mayor. Esta información sólo refleja la actividad generada por los usuarios monitoreados que no están asociados a ningún perfil profesional en particular.

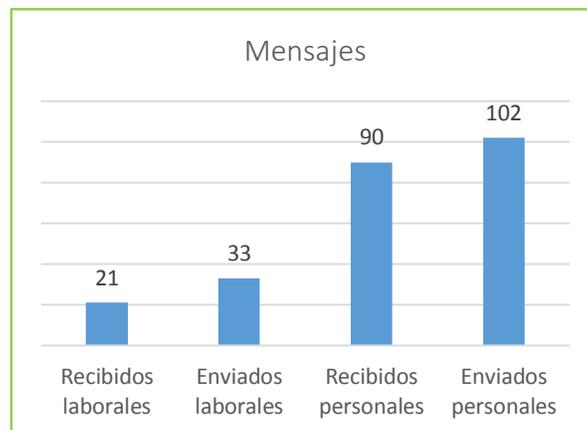


Figura 5.7 Actividad de las variables de la tecnología mensajes

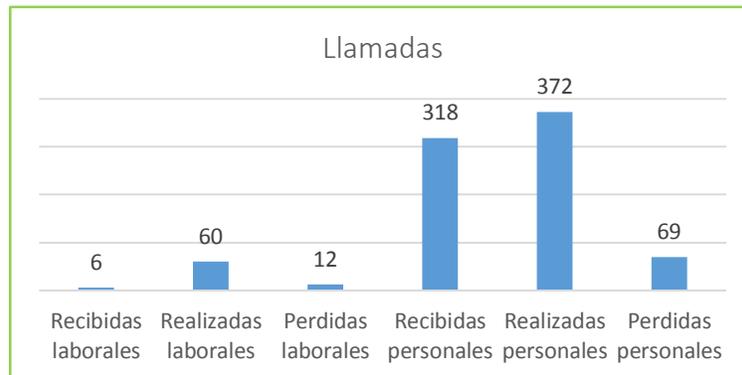


Figura 5.8 Actividad de las variables de la tecnología llamadas

La aplicación web fue consultada en los navegadores Chrome, Firefox, Internet Explorer y Android browser, los cuales permitieron mostrar una página web con gráficas que reflejan la información completa que recupera el sistema "LaborCheck". Es decir, se mostraron las interacciones del teclado, interacciones del mouse, interacciones con ventanas, registros de mensajes y registros de llamadas. Estas gráficas permitieron que el usuario observara su información de una forma interactiva debido a que la librería utilizada permite realizar una selección por variables para poder realizar un análisis visual de los datos.

La aplicación usuario-computadora utiliza 24 MB de memoria RAM. Las aplicaciones relacionadas utilizan 10 MB, sin embargo, son aplicaciones que monitorean cada tecnología en aplicaciones separadas, mientras que LaborCheck monitorea todas las tecnologías en una sola aplicación.

La aplicación de interacción social utiliza 4.7 MB de memoria RAM que en comparación con la aplicación relacionada es mínima puesto que la aplicación relacionada utiliza 18 MB de memoria para su ejecución.

El espacio de almacenamiento en la base de datos de las tecnologías fue de 1218 KB para teclado, 1349 KB para mouse, 976 KB para ventanas, 582 KB para mensajes y 657 KB para llamadas. Por lo tanto, las variables relacionadas con el mouse almacenaron la mayor cantidad de información mientras que la tecnología mensajes almacenó la mínima cantidad.

Capítulo 6

Conclusiones y Trabajos Futuros

Capítulo 6. Conclusiones y Trabajos Futuros

En este capítulo se describen las conclusiones en donde se mencionan las principales aportaciones tecnológicas generadas con este trabajo de investigación. También se nombran los logros que se obtuvieron durante la formación como investigador. Finalmente, se describen los trabajos de investigación que se pueden generar a partir de las tecnologías incluidas en el sistema LaborCheck.

6.1 Conclusiones

En este trabajo de investigación se recuperaron un total de 23 variables por medio de dos aplicaciones de monitoreo no intrusivas que recuperan los valores de las variables de forma automática. La primera aplicación recupera interacciones usuario-computadora y se ejecuta en el sistema operativo Windows. La segunda aplicación recupera interacciones de comunicación social por medio de mensajes y llamadas del *Smartphone* y se ejecuta en el sistema operativo Android.

Los valores de las variables que se recuperan, se muestran al usuario por medio de una aplicación que puede ser ejecutada en cualquier navegador web. La aplicación web se adapta al tamaño del dispositivo en el que se esté visualizando. Por lo tanto, el usuario puede consultar la información que es mostrada por medio de gráficas de series tanto en equipos de escritorio como en dispositivos móviles sin importar el sistema operativo.

Por medio de la tecnología relacionada con el teclado se recuperaron automáticamente los valores de las variables que pueden estar asociadas al comportamiento que representa niveles de estrés laboral. Con esta tecnología se clasificaron diferentes tipos de teclas que son contabilizadas para que en un futuro se analice esta información como teclas de error y otras teclas que se presionan frecuentemente como la barra espaciadora.

Respecto a la tecnología relacionada con el mouse se recolectaron automáticamente los valores de las variables obtenidas con los componentes del mouse. Los resultados presentados con el uso del mouse mostraron que este dispositivo presenta una mayor actividad respecto a las otras tecnologías. Esta información será útil para comprobar cómo se comportan los usuarios que presentan episodios de estrés al hacer uso de esta tecnología.

Con la tecnología relacionada con ventanas de Windows se recuperaron correctamente los movimientos generados con las ventanas. Con estos cambios se podrá analizar el comportamiento de estas variables para detectar situaciones de estrés laboral. Los cambios entre ventanas se detectan cuando se generan por medio de clics entre ventanas o con la combinación de teclas control más tabulador.

Las tecnologías relacionadas con mensajes y llamadas permitieron detectar la actividad de comunicación entre el usuario y los contactos que se agregaron en el módulo de la aplicación de interacción social. Estas tecnologías presentaron menor actividad que las interacciones usuario-computadora. Sin embargo, se pudo observar que la mayor actividad generada con estas tecnologías se presentó cuando a los usuarios se les negó el acceso a redes sociales y otros medios que utilizan internet como medio de comunicación. Por lo tanto, estos valores se

incrementarán en situaciones similares cuando los empleados sólo tengan la opción de comunicarse por medio de mensajes y llamadas.

El reinicio automático tanto de la aplicación de interacción usuario-computadora como de la aplicación de interacción social permitió que los usuarios en ciertas ocasiones se olvidaran de que su actividad estaba siendo monitoreada. Por lo tanto, se comprobó que las aplicaciones desarrolladas no son intrusivas para los usuarios y pueden permanecer instaladas en los dispositivos de los usuarios. Esto permitirá prolongar los periodos de tiempo para la recolección de datos.

El sistema LaborCheck también almacenó los periodos en los que el usuario no realiza ninguna actividad con estas tecnologías. Esto permite contabilizar el tiempo en el que un empleado podría estar realizando otras actividades que le permitan disminuir la tensión generada con el uso de estas tecnologías. Al mismo tiempo puede identificar los lapsos prolongados de tiempo en los que se encuentra utilizando estas tecnologías que podrían llevarlo a presentar situaciones de estrés laboral.

En comparación con el software relacionado, LaborCheck únicamente almacena las cantidades que se recuperan con estas variables, no almacena información que es privada para los usuarios. Por lo tanto, el usuario puede sentirse seguro al saber que con las aplicaciones de monitoreo del sistema LaborCheck ninguna información privada que se captura por medio del teclado, mensajes y llamadas es almacenada.

Con el uso de las bases de datos locales los valores de las interacciones se conservan en el disco duro del dispositivo con el que se esté monitoreando. Con esto se asegura que la información no se pierda antes de enviarse al servidor remoto porque en algunas ocasiones los dispositivos con los que se está monitoreando podrían no tener acceso a internet.

El sistema LaborCheck permite que los desarrolladores incrementen la funcionalidad de las aplicaciones fácilmente debido a que se estableció una programación basada en el modelo vista controlador. Esta programación facilita que los códigos sean modificados de forma independiente en cada capa del modelo.

Este sistema distribuido almacena los valores recuperados por las aplicaciones de monitoreo en una base de datos centralizada. La información almacenada de todas las tecnologías se podrá utilizar para analizarla y detectar en tiempo real periodos de estrés y otros problemas que se relacionen con estas tecnologías.

Con este tipo de aplicaciones de monitoreo en tiempo real se puede obtener una gran cantidad de información de múltiples usuarios simultáneamente debido a que estas aplicaciones se configuran fácilmente y son intuitivas para el usuario.

Con la aplicación web los usuarios pueden visualizar fácilmente su actividad computacional y móvil. De esta forma los usuarios podrán ser conscientes de la actividad generada con el uso de estas tecnologías y crear hábitos saludables que reduzcan problemas de salud como el estrés laboral.

6.2 Logros obtenidos

- Participación en el proyecto “Emotion-Bracelet”, presentado en el Evento Nacional de Innovación Tecnológica en su Etapa Local, celebrado el 12 de junio del 2015 en las instalaciones del Centro Nacional de Investigación y Desarrollo Tecnológico.
- Aceptación del artículo “LaborCheck: Sistema de Monitoreo Automático de Variables Usuario-Computadora y de Interacción Social” en el taller de computación clínica e informática médica en el marco del encuentro nacional en computación presentado en el estado de Chihuahua en noviembre del 2016.

6.3 Trabajos futuros

Los niveles de estrés laboral se podrán clasificar analizando la información obtenida de las variables con métodos estadísticos y modelos de aprendizaje automático. Por lo tanto, se podrán incorporar mensajes o algún tipo de alerta sobre las aplicaciones de monitoreo que ayuden a reducir el estrés laboral detectado con el uso de estas tecnologías.

Algunas notificaciones que se pueden incorporar en la aplicación de interacción usuario-computadora son mensajes de alerta cuando se detecte que los empleados han pasado largos periodos de tiempo utilizando el teclado, mouse y generando movimientos con las ventanas de Windows.

La productividad será posible detectarla por medio las interacciones usuario-computadora, especialmente analizando el total de teclas presionadas contra las teclas de tipo error como suprimir o retroceso, así como los tiempos de actividad y de inactividad. Adicionalmente se puede extender el monitoreo de las teclas que se utilizan para copiar y pegar texto, y asociarlas con la productividad.

Los problemas relacionados con articulaciones causados principalmente en las manos, brazos, o cuello se podrán detectar por medio de las interacciones usuario-computadora, debido a que el uso excesivo de estas tecnologías comúnmente provoca tensión en diferentes zonas del cuerpo.

Los problemas sociales que existen en nuestro entorno como soledad, depresión, aislamiento, etcétera., se podrán detectar utilizando la información relacionada con la comunicación social por medio de mensajes y llamadas. Esta información se podrá completar extrayendo otra información social obtenida de aplicaciones como WhatsApp o Facebook.

Las aplicaciones de monitoreo se podrán modificar para incorporar formularios que permitan personalizar los lapsos de tiempo para recuperan los valores de las variables. De esta manera se podrá analizar si entre mayores o menores son los lapsos de tiempo se pueden obtener mejores patrones de comportamiento sobre diferentes aplicaciones.

El interés de los usuarios sobre las páginas web que visitan se podrá deducir por medio de las interacciones generadas con el mouse y el teclado. En la aplicación web se podrán agregar los temas de interés de cada usuario y con ellos formar un perfil acorde a las interacciones generadas sobre las páginas web que se visitan.

Los estados emocionales que se obtuvieron en otros artículos utilizando únicamente la dinámica del teclado se podrán obtener al anexas la tecnología del mouse y ventanas.

Respecto a la identidad del usuario que se comparó con la dinámica del teclado ahora se podrán incorporar las tecnologías mouse y ventanas para incrementar el reconocimiento del usuario por medio de estas tres tecnologías.

Con el fin de ampliar la recolección de interacciones que genera el usuario ya sea entre la computadora o con los dispositivos móviles, otras aplicaciones se pueden desarrollar para ampliar la extracción de variables dentro del sistema LaborCheck. Por ejemplo:

- Una aplicación que contabilice la cantidad de pulsaciones generadas sobre el teclado de los dispositivos móviles.
- Una aplicación que contabilice la cantidad de pestañas abiertas de los navegadores web y que sea compatible con los diferentes navegadores: Chrome, Internet Explorer, Mozilla Firefox, etcétera.
- Una aplicación que obtenga las interacciones generadas por medio de otras plataformas como Linux y Mac. Estas aplicaciones podrían conectarse con la base de datos centralizada del sistema LaborCheck, con lo cual se ampliaría la cantidad de usuarios que podrían ser monitoreados y comprobar si existen algunas variaciones entre el uso de estas tecnologías con diferentes plataformas.
- El módulo de contactos permitió que se clasificaran los contactos del usuario como personales o laborales. Sin embargo, este módulo podrá ser modificado para clasificar los contactos en las categorías que se adapten a las necesidades de cada aplicación. Por ejemplo, en aplicaciones de soledad y aislamiento pueden ser contactos familiares, contactos de amigos, contactos por grupo social etc.

La base de datos del sistema LaborCheck se podrá convertir en un repositorio que almacene la información de diferentes instituciones de investigación u otras organizaciones y así poder comparar las interacciones generadas entre instituciones, edades, y perfiles de usuarios.

Una nueva red social en donde los usuarios publiquen la actividad generada durante el día se podrá desarrollar para compartir los diferentes niveles de estrés o diferentes emociones obtenidas por medio de la actividad computacional y móvil. En esta red social los usuarios podrían mejorar su desempeño al permitir que se publiquen sólo mensajes positivos en los comentarios.

Las palabras completas se podrán analizar con técnicas de lenguaje natural si se modifica la aplicación que utiliza la tecnología teclado para almacenar estas palabras localmente en el equipo de los usuarios. Por medio de estas técnicas se podrán detectar en tiempo real las emociones que se reflejan en los documentos escritos en diferentes programas o correos electrónicos.

Las coordenadas del mouse se podrán almacenar al modificar la aplicación que recupera los valores de la tecnología mouse y utilizarlas en diferentes programas que permitan analizar el seguimiento de mouse en la toma de decisiones.

Diferentes interacciones se pueden incorporar al sistema LaborCheck para utilizarlas en varias aplicaciones. Sin embargo, es importante realizar un análisis para decidir qué tipos de variables se pueden asociar a los diferentes problemas sociales o enfoques de investigación.

Referencias

- [1] Global Organization for Stress, "STRESS RELATED FACTS AND STATISTICS." [Online]. Available: <http://www.gostress.com/stress-facts/>. [Accessed: 13-Aug-2015].
- [2] El Universal, "México, país con más estrés laboral." [Online]. Available: <http://www.eluniversal.com.mx/ciencia/2013/mexico-pais-mas-estres-laboral-78744.html>. [Accessed: 24-Mar-2015].
- [3] N. Sharma and T. Gedeon, "Objective measures, sensors and computational techniques for stress recognition and classification: A survey," *Comput. Methods Programs Biomed.*, vol. 108, no. 3, pp. 1287–1301, 2012.
- [4] G. Pulido, "El Estrés." [Online]. Available: <http://www.insp.mx/cuidando-tu-salud/1386-el-estres-ique-es-y-com>. [Accessed: 14-Mar-2015].
- [5] Organización Mundial de la Salud, Salud Mental y Abuso de Sustancias, "Prevención del suicidio, un instrumento en el trabajo," 2006. [Online]. Available: http://www.who.int/mental_health/prevention/suicide/resource_work_spanish.pdf. [Accessed: 11-May-2015].
- [6] Organización Internacional del Trabajo, "Evaluación de impacto." [Online]. Available: <http://guia.oitcinterfor.org/conceptualizacion/diferencia-monitoreo-intermedia-evaluacion-impacto>. [Accessed: 15-Jun-2016].
- [7] Organisation for Economic Co-operation and Development, "Balance vida-trabajo," 2015. [Online]. Available: <http://www.oecdbetterlifeindex.org/es/countries/mexico-es/>. [Accessed: 16-Mar-2015].
- [8] Organización Mundial de la Salud and Organización Panamericana de la Salud, "Ambientes de trabajo saludables: un modelo para la acción," 2010. [Online]. Available: http://www.who.int/occupational_health/healthy_workplaces_spanish.pdf. [Accessed: 15-May-2015].
- [9] Z. Hinbarji, R. Albatat, N. O 'connor, and C. Gurrin, "LoggerMan, A Comprehensive Logging and Visualization Tool to Capture Computer Usage," in *MultiMedia Modeling: 22nd International Conference, MMM 2016, Miami, FL, USA, January 4-6, 2016, Proceedings, Part II*, Q. Tian, N. Sebe, G.-J. Qi, B. Huet, R. Hong, and X. Liu, Eds. Cham: Springer International Publishing, 2016, pp. 342–347.
- [10] J. B. Freeman and N. Ambady, "MouseTracker: Software for studying real-time mental processing using a computer mouse-tracking method," *Behav. Res. Methods*, vol. 42, no. 1, pp. 226–241, 2010.
- [11] C. Epp, M. Lippold, and R. L. Mandryk, "Identifying Emotional States Using Keystroke Dynamics," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2011, pp. 715–724.
- [12] V. Deufemia, M. Giordano, G. Polese, and G. Tortora, "Capturing User's Interest from Human-Computer Interaction Logging," in *Web Information Systems and Technologies: 8th International Conference, WEBIST 2012, Porto, Portugal, April 18-21, 2012, Revised Selected Papers*, J. Cordeiro and K.-H. Krempels, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 312–327.
- [13] T. Stütz, T. Kowar, M. Kager, M. Tiefengrabner, M. Stuppner, J. Blechert, F. H. Wilhelm, and S. Ginzinger, "Smartphone Based Stress Prediction," in *User Modeling, Adaptation and*

- Personalization: 23rd International Conference, UMAP 2015, Dublin, Ireland, June 29 -- July 3, 2015. Proceedings*, F. Ricci, K. Bontcheva, O. Conlan, and S. Lawless, Eds. Cham: Springer International Publishing, 2015, pp. 240–251.
- [14] R. Mafrur, I. G. D. Nugraha, and D. Choi, "Modeling and discovering human behavior from smartphone sensing life-log data for identification purpose," *Human-centric Comput. Inf. Sci.*, vol. 5, no. 1, pp. 1–18, 2015.
- [15] Instituto Nacional para la Seguridad y Salud Ocupacional, "El estrés en el trabajo," 99-101, 1999. [Online]. Available: http://www.cdc.gov/spanish/niosh/docs/99-101_sp/. [Accessed: 15-Mar-2015].
- [16] Organización Mundial de la Salud, Houtman Irene, "Sensibilizando sobre el Estrés Laboral en los Países en Desarrollo," *Protección de la Salud de los Trabajadores*, 2008. [Online]. Available: http://whqlibdoc.who.int/publications/2008/9789243591650_spa.pdf. [Accessed: 14-Apr-2015].
- [17] P. Montuschi, A. Sanna, F. Lamberti, and G. Paravati, "Interacción entre Humanos y Computadoras: Tendencias Actuales y Futuras," 2014. [Online]. Available: <http://www.computer.org/web/computingnow/archive/september2014-spanish>. [Accessed: 20-Nov-2015].
- [18] Techopedia Inc., "Human-Computer Interaction (HCI)," 2016. [Online]. Available: <https://www.techopedia.com/definition/3639/human-computer-interaction-hci>. [Accessed: 20-Jun-2016].
- [19] Microsoft, "Introducción a .NET Framework," 2016. [Online]. Available: [https://msdn.microsoft.com/es-mx/library/hh425099\(v=vs.110\).aspx](https://msdn.microsoft.com/es-mx/library/hh425099(v=vs.110).aspx). [Accessed: 21-Jun-2016].
- [20] Microsoft, "Hooks," 2016. [Online]. Available: [https://msdn.microsoft.com/en-us/library/windows/desktop/ms644959\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms644959(v=vs.85).aspx). [Accessed: 21-Jun-2016].
- [21] Microsoft, "BackgroundWorker Class," 2016. [Online]. Available: [https://msdn.microsoft.com/en-us/library/system.componentmodel.backgroundworker\(v=vs.95\).aspx](https://msdn.microsoft.com/en-us/library/system.componentmodel.backgroundworker(v=vs.95).aspx). [Accessed: 15-Jun-2016].
- [22] M. Báez, Á. Borrego, J. Cordero, and L. Cruz, "Introducción a Android," *Ceavitoria.Com.Es*, p. 1, 2012.
- [23] W. Service, S. Object, A. Protocol, M. Soap, X. Participante, J. M. S. Una, and J. M. S. B. Binding, "Servicios web." [Online]. Available: <http://itpn.mx/recursosisc/8semestre/programacionweb/Unidad VI.pdf>. [Accessed: 21-Nov-2015].
- [24] W3C, "Simple Object Access Protocol (SOAP) 1.1," 2000. [Online]. Available: https://www.w3.org/TR/2000/NOTE-SOAP-20000508/#_Toc478383489. [Accessed: 21-Jun-2016].
- [25] ECMA-404, "The JSON Data Interchange Format," *ECMA Int.*, vol. 1st Editio, no. October, p. 8, 2013.
- [26] W3C, "Tecnologías XML," 2016. [Online]. Available: <http://www.w3c.es/Divulgacion/GuiasBreves/TecnologiasXML>. [Accessed: 10-Jun-2016].
- [27] Universidad de las Américas de Puebla, "Cliente-Servidor," 2016. [Online]. Available: http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/marquez_a_bm/capitulo5.pdf. [Accessed: 14-Jun-2016].

-
- [28] D. G. Y. Fernández Romero Yenisleidy, "Patrón Modelo-Vista-Controlador," *Rev. Telemática*, vol. 11, no. 1, p. 11, 2012.
- [29] A. (Bell L. Silberschatz, H. F. (Bell L. Korth, and B. Sudarshan, S. (Instituto Indio de Tecnología, *Fundamentos de bases de datos*. 2002.
- [30] Hipp, Wyrick & Company, Inc., "SQLite," 2016. [Online]. Available: <https://www.sqlite.org/about.html>. [Accessed: 10-Apr-2016].
- [31] W. Dubitzky, O. Wolkenhauer, K.-H. Cho, and H. Yokota, Eds., "MySQL: My Structured Query Language," in *Encyclopedia of Systems Biology*, New York, NY: Springer New York, 2013, pp. 1485–1486.
- [32] Universitat Oberta de Catalunya, C. M. Escofet, "El lenguaje sql," *Uoc*, pp. 1–42, 2007.
- [33] Actual Keylogger Software Inc., "Actual Keylogger," 2016. [Online]. Available: <http://www.actualkeylogger.com/>. [Accessed: 08-Jun-2016].
- [34] Bartels Media GmbH Company, "Mouse Recorder," 2016. [Online]. Available: <http://www.mouserecorder.com/>. [Accessed: 09-Jun-2016].
- [35] Refog Inc., "Hoverwatch," 2016. [Online]. Available: <https://www.hoverwatch.com/es/>. [Accessed: 12-Jun-2016].

Anexos

Anexo A. Descripción de la base de datos local de la aplicación de interacción usuario-computadora

Tabla propiedades. En esta tabla se almacena la información relacionada con el usuario que inicia la aplicación de interacción usuario-computadora.

- *id*. En este campo se almacena el identificador de esta tabla.
- *id_usuario*. En este campo se almacena el *id* que idéntica al usuario, este *id* es recuperado de la base de datos remota después de que se inicia sesión en la aplicación de interacción usuario-computadora y posteriormente es almacenado en esta tabla.

Tabla teclado. En esta tabla se almacenan los valores recuperados de las interacciones generadas con el teclado, los cuales son almacenados con un lapso de tiempo de cinco minutos.

- *id*. En este campo se almacena el identificador de esta tabla.
- *total_teclas*. En este campo se almacena la cantidad total de teclas presionadas.
- *total_backspace*. En este campo se almacena la cantidad de pulsaciones generadas sobre la tecla retroceso.
- *total_delete*. En este campo se almacena la cantidad de pulsaciones generadas sobre la tecla eliminar.
- *total_enter*. En este campo se almacena la cantidad de pulsaciones generadas sobre la tecla de tipo enter.
- *total_spacebar*. En este campo se almacena la cantidad de pulsaciones generadas sobre la tecla espaciadora.
- *fecha_hora*. En este campo se almacena la fecha y hora obtenida respecto al lapso de tiempo en el que se recuperan estas interacciones.
- *id_usuario*. En este campo se almacena el *id* del usuario que está relacionado con la tabla propiedades, lo cual permite identificar que estas interacciones correspondan con el usuario autenticado.

Tabla mouse. En esta tabla se almacenan los valores recuperados de las interacciones generadas con el mouse. Estos valores se almacenan en esta tabla cada cinco minutos.

- *id*. En este campo se almacena el identificador de la tabla.
- *total_clics*. En este campo se almacena la cantidad de clics presionados de cualquier tipo.
- *total_clics_izq*. En este campo se almacena la cantidad de clics presionados con el botón izquierdo.
- *total_clics_der*. En este campo se almacena la cantidad de clics presionados con el botón derecho.
- *total_scroll_arr*. En este campo se almacena la cantidad de desplazamientos generados hacia arriba con la rueda del mouse.

- *total_scroll_ab*. En este campo se almacena la cantidad de desplazamientos generados hacia abajo con la rueda del mouse.
- *total_pixeles*. En este campo se almacena la cantidad de pixeles recorridos sobre la pantalla con el puntero del mouse.
- *fecha_hora*. En este campo se almacena la fecha y hora que corresponde al lapso de tiempo en el que se recuperan las interacciones del mouse.
- *id_usuario*. En este campo se almacena el identificador del usuario que corresponde al usuario que genera las interacciones.

Tabla ventana. En esta tabla se almacenan los valores recuperados de las interacciones generadas con las ventanas de Windows, los cuales se almacenan con un lapso de tiempo de cinco minutos.

- *id*. En este campo se almacena el identificador de la tabla.
- *total_ventanas_abiertas*. En este campo se almacena la cantidad de ventanas que abre el usuario mientras se encuentra trabajando con su computadora.
- *total_cambios*. En este campo se almacena la cantidad de cambios de ventana que genera el usuario. También se detectan los cambios entre ventanas generados con la combinación de teclas control más tabulador.
- *fechaHora*. En este campo se almacena la fecha y hora que corresponde al lapso de tiempo en el que se almacenan estas interacciones, el cual es registrado cada cinco minutos.
- *id_usuario*. En este campo se almacena el identificador que corresponde al usuario que está siendo monitoreado.

Anexo B. Descripción de la base de datos local de la aplicación de interacción social

Tabla propiedades. En esta tabla se almacena la información del usuario que inicia sesión dentro de la aplicación de interacción social.

- *id*. En este campo se almacena el identificador de la tabla.
- *id_usuario*. En este campo se guarda el *id* del usuario, el cual es obtenido después de haber validado que los datos de inicio de sesión sean correctos. Este *id* identifica al usuario después de que el dispositivo móvil se reinicia.
- *usuario*. En este campo se almacena el nombre del usuario que inicia sesión en la aplicación de interacción social.

Tabla contactos. En esta tabla se almacena la información relacionada con los contactos del usuario.

- *id*. En este campo se almacena el identificador de la tabla.
- *id_tipo_contacto*. En este campo se almacena el tipo de contacto que es asignado por el usuario al momento de registrarlo, los valores pueden ser: 1, que indica que es un contacto laboral y 2, que indica que es un contacto personal.
- *numero_telefono*. En este campo se almacena el número de teléfono del contacto.
- *nombre*. En este campo se almacena el nombre del nuevo contacto.
- *id_usuario*. En este campo se almacena el *id* del usuario al que pertenecen los contactos.

Tabla conteomensajes. En esta tabla se almacena la cantidad de mensajes que son detectados por la aplicación. El valor de cada campo está relacionado con el tipo de mensaje detectado y el tipo de contacto asociado al mensaje. Este valor se incrementa en uno cada vez que se detecta un nuevo mensaje. Inicialmente el valor de estos campos se inserta con un valor de cero.

- *id*. En este campo se almacena el identificador de la tabla.
- *msgs_rec_lab*. En este campo se incrementa el valor de los mensajes recibidos laborales.
- *msgs_rec_per*. En este campo se incrementa el valor de los mensajes recibidos personales.
- *msgs_env_lab*. En este campo se incrementa el valor de los mensajes enviados laborales.
- *msgs_env_per*. En este campo se incrementa el valor de los mensajes enviados personales.
- *id_ultimo_mensaje*. En este campo se almacena el identificador del último mensaje detectado, este identificador se compara con el identificador del nuevo mensaje para incrementarlo en el campo que corresponda al tipo de mensaje y tipo de contacto.
- *id_usuario*. En este campo se almacena el identificador del usuario que asocia los valores registrados con el usuario que está siendo monitoreado.

Tabla conteollamadas. En esta tabla se almacena la cantidad de llamadas que son detectadas por la aplicación. El valor de cada campo está relacionado con el tipo de llamadas detectadas y el tipo de contacto asociado a las llamadas. Este valor se incrementa en uno cada vez que se detecta un registro de llamada. Inicialmente el valor de estos campos se registra con un valor de cero.

- *id*. En este campo se almacena el identificador de la tabla.
- *calls_sal_lab*. En este campo se incrementa el valor de las llamadas realizadas laborales.
- *calls_sal_per*. En este campo se incrementa el valor de las llamadas realizadas personales.
- *calls_ent_lab*. En este campo se incrementa el valor de las llamadas entrantes laborales.
- *calls_ent_per*. En este campo se incrementa el valor de las llamadas entrantes personales.
- *calls_perd_lab*. En este campo se incrementa el valor de las llamadas perdidas laborales.
- *calls_perd_per*. En este campo se incrementa el valor de las llamadas perdidas personales.
- *id_ultima_llamada*. En este campo se almacena el identificador de la última llamada detectada, el cual se compara con el identificador de la última llamada registrada para que en caso de ser diferente se incremente el valor en el campo que corresponda al tipo de llamada y tipo de contacto.
- *id_usuario*. En este campo se almacena el *id* del usuario para asociar los valores de las llamadas con el usuario que está siendo monitoreado.

Tabla registro_mensajes. En esta tabla se almacenan los valores recuperados de la tabla conteomensajes con un lapso de tiempo de dos horas.

- *id*. En este campo se almacena el identificador de la tabla.
- *total_rec_lab*. En este campo se almacenan los valores de los mensajes recibidos laborales.
- *total_rec_per*. En este campo se almacenan los valores de los mensajes recibidos personales.
- *total_env_lab*. En este campo se almacenan los valores de los mensajes enviados laborales.
- *total_env_per*. En este campo se almacenan los valores de los mensajes enviados personales.
- *fecha_hora*. En este campo se almacena la fecha y hora de registro de los valores de esta tabla, la cual coincide con el lapso de tiempo de dos horas.
- *id_usuario*. En este campo se almacena el identificador del usuario para asociar estos valores recuperados de cada dos horas con el usuario que se está monitoreando.

Tabla registro_llamadas. En esta tabla se almacenan los valores recuperados de la tabla conteollamadas con un lapso de tiempo de dos horas.

- *id.* En este campo se almacena el identificador de la tabla.
- *total_sal_lab.* En este campo se almacenan los valores de las llamadas realizadas laborales.
- *total_sal_per.* En este campo se almacenan los valores de las llamadas realizadas personales.
- *total_ent_lab.* En este campo se almacenan los valores de las llamadas entrantes laborales.
- *total_ent_per.* En este campo se almacenan los valores de las llamadas entrantes personales.
- *total_perd_lab.* En este campo se almacenan los valores de las llamadas perdidas laborales.
- *total_perd_per.* En este campo se almacenan los valores de las llamadas perdidas personales.
- *fecha_hora.* En este campo se almacena la fecha y hora en la que se registraron los valores de esta tabla, la cual coincide con el lapso de tiempo de dos horas.
- *id_usuario.* En este campo se almacena el identificador del usuario registrado para asociar los valores de esta tabla con el usuario que está siendo monitoreado por la aplicación.

Anexo C. Descripción de la base de datos remota del sistema LaborCheck

Tabla usuarios. En esta tabla se almacena la información de cada usuario. Las tres aplicaciones incluidas en el sistema LaborCheck tienen acceso a esta tabla ya sea para insertar nuevos usuarios o para autenticar.

- *id_usuario.* En este campo se almacena el identificador del registro por medio de un campo auto-incremental.
- *usuario.* En este campo se almacena el nombre de usuario.
- *contrasena.* En este campo se almacena la contraseña proporcionada por el usuario.

Tabla usuario_tecnologias. En esta tabla se almacenan las tecnologías pertenecientes a cada usuario. Las tecnologías que monitorean las interacciones usuario-computadora son insertadas desde la aplicación Windows mientras que las tecnologías que monitorean las interacciones generadas por comunicación son insertadas desde la aplicación instalada en el sistema operativo Android.

- *id_usuario_tecnologia.* En este campo se almacena el identificador único de la tabla.
- *id_tecnologia.* En este campo se almacena el *id* que se relaciona con la tabla que contiene el nombre de la tecnología que se monitorea.
- *id_usuario.* En ese campo se almacena el *id* que se relaciona con el usuario que está siendo monitoreado.

Tabla tecnologias. En esta tabla se almacenan los nombres de las tecnologías incluidas en el sistema de monitoreo: mouse, teclado, ventanas, mensajes y llamadas.

- *id_tecnologia.* En este campo se almacena el identificador único de la tabla.
- *nombre.* En este campo se almacena el nombre de la tecnología.

Tabla registros_teclado. En esta tabla se almacena la información de los registros del teclado que son obtenidos cada cinco minutos por la aplicación de interacción usuario-computadora.

- *id_registro_teclado.* En este campo se almacena el identificador de la entidad.
- *total_teclas.* En este campo se almacena la cantidad de teclas presionadas.
- *total_backspace.* En este campo se almacena la cantidad de teclas tipo retroceso.
- *total_delete.* En este campo se almacena la cantidad de teclas tipo eliminar.
- *total_enter.* En este campo se almacena la cantidad de teclas tipo enter.
- *total_spacebar.* En este campo se almacena la cantidad de teclas presionadas con la barra espaciadora.
- *fecha_hora.* En este campo se almacena la fecha y hora de registro.
- *id_usuario_tecnologia.* En este campo se almacena el *id* que se relaciona con la tabla *usuario_tecnologias*.

Tabla registros_mouse. En esta tabla se almacena la información de los registros del mouse registrados cada cinco minutos por la aplicación de interacción usuario-computadora.

- *id_registro_mouse.* En este campo se almacena el identificador de la tabla.
- *total_clics.* En este campo se almacena la cantidad de clics presionados con cualquier botón del mouse.
- *total_clics_izq.* En este campo se almacena la cantidad de clics presionados con el botón izquierdo.
- *total_clics_der.* En este campo se almacena la cantidad de clics presionados con el botón derecho.
- *total_scroll_arr.* En este campo se almacena la cantidad de desplazamientos hacia arriba generados con la rueda del mouse.
- *total_scroll_ab.* En este campo se almacena la cantidad de desplazamientos hacia abajo generados con la rueda del mouse.
- *total_pixeles.* En este campo se almacena la cantidad de pixeles recorridos con el puntero del mouse sobre la pantalla.
- *fecha_hora.* En este campo se almacena la fecha y hora del registro.
- *id_usuario_tecnologia.* En este campo se almacena el identificador que se relaciona con la tabla usuario_tecnologias.

Tabla registros_ventanas. En esta tabla se almacena la información de los registros de ventanas almacenados cada cinco minutos por la aplicación de interacción usuario-computadora.

- *id_registro_ventana.* En este campo se almacena el identificador del registro.
- *total_ventanas_abiertas.* En este campo se almacena la cantidad de ventanas abiertas.
- *total_cambios.* En este campo se almacena la cantidad de cambios entre ventanas.
- *fechaHora.* En este campo se almacena la fecha y hora del registro.
- *id_usuario_tecnologia.* En este campo se almacena el *id* que se relaciona con la tabla usuario_tecnologias.

Tabla registros_mensajes. En esta tabla se almacena la información de los registros de mensajes almacenados cada dos horas por la aplicación de interacción social.

- *id_registro_mensaje.* En este campo se almacena el identificador de la tabla.
- *msj_ent_trab.* En este campo se almacena la cantidad de mensajes recibidos laborales.
- *msj_sal_trab.* En este campo se almacena la cantidad de mensajes enviados laborales.
- *msj_ent_pers.* En este campo se almacena la cantidad de mensajes recibidos personales.
- *msj_sal_pers.* En este campo se almacena la cantidad de mensajes enviados personales.
- *fecha_hora.* En este campo se almacena la fecha y hora del registro.
- *id_usuario_tecnologia.* En esta tabla se almacena el *id* de la tabla usuario_tecnologias.

Tabla registros_llamadas. En esta tabla se almacena la información de los registros de llamadas almacenados cada dos horas por la aplicación de interacción social.

- *id_registro_llamada.* En este campo se almacena el identificador de la entidad.
- *llam_ent_trab.* En este campo se almacena la cantidad de llamadas entrantes laborales.
- *llam_sal_trab.* En este campo se almacena la cantidad de llamadas realizadas laborales.
- *llam_per_trab.* En este campo se almacena la cantidad de llamadas perdidas laborales.
- *llam_ent_pers.* En este campo se almacena la cantidad de llamadas entrantes personales.
- *llam_sal_pers.* En este campo se almacena la cantidad de llamadas realizadas personales.
- *llam_per_pers.* En este campo se almacena la cantidad de llamadas perdidas personales.
- *fecha_hora.* En este campo se almacena la fecha y hora del registro.
- *id_usuario_tecnologia.* En este campo se almacena el *id* de la tabla usuario_tecnologias.

Anexo D. Casos de prueba para la aplicación de interacción usuario-computadora

CP_AUC_1

Aprobado

Desarrollo de la prueba:

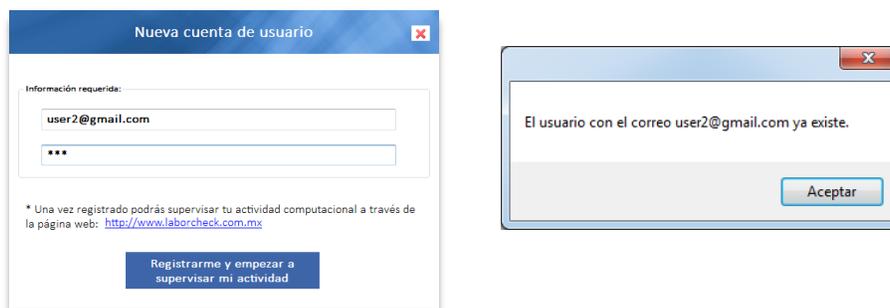
1. Dar clic sobre la opción "crear nueva cuenta de usuario", esta opción se muestra en el formulario de inicio de sesión. Dar clic sobre el botón "Registrarme y empezar a supervisar mi actividad" con los campos vacíos del formulario. La aplicación mostrará un mensaje de validación notificando que se ingresen los datos necesarios para el registro: nombre de usuario y contraseña.
2. Ingresar un nombre de usuario que este almacenado en la base de datos. La aplicación deberá mostrar un mensaje de validación notificando que el usuario con el nombre de usuario ya ha sido ingresado anteriormente debido a que este campo se comporta como la clave única del usuario.
3. Ingresar información que no se encuentren almacenada en la base de datos. Una vez ingresados estos datos, la aplicación deberá minimizarse en la barra de tareas.
4. Dar clic derecho sobre el icono minimizado de la aplicación. Se deberá mostrar un enlace con el siguiente mensaje: "Clic para supervisar mi actividad en la página web: <http://www.laborcheck.com.mx>", esta opción abrirá la aplicación web de consulta de información, el contenido de esta aplicación se explica con más detalle en el plan de pruebas de la aplicación web.

Evidencias del desarrollo de la prueba:

1.



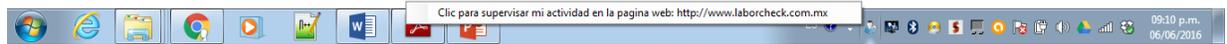
2.



3.



4.



Observaciones: La aplicación validó correctamente los campos requeridos, evitando que se registrara el mismo usuario más de una vez en la base de datos remota. Los datos de registro se almacenaron correctamente y la aplicación se minimizó en el área de notificación de la barra de tareas.

CP_AUC_2

Aprobado

Desarrollo de la prueba:

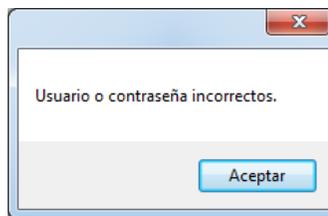
1. Dar clic sobre el botón "inicia sesión" con los campos vacíos en el formulario. La aplicación deberá mostrar un mensaje de validación solicitando los datos requeridos.
2. Capturar un nombre de usuario y una contraseña erróneos. La aplicación deberá mostrar un mensaje de validación notificando que los datos con los que se intenta iniciar sesión son incorrectos, es decir, no están registrados en la base de datos.
3. Ingresar los datos que se encuentren registrados en la base de datos remota y finalmente dar clic en el botón "inicia sesión". La aplicación no deberá mostrar mensajes de validación y procederá a minimizarse en la barra de tareas.
4. Dar clic derecho sobre el icono de la aplicación minimizada en la barra de tareas. La aplicación deberá mostrar un enlace con el siguiente mensaje: "Clic para supervisar mi actividad en la página web: <http://www.laborcheck.com.mx>", esta opción abrirá la aplicación web de consulta de información, el contenido de esta aplicación se explica con más detalle en el plan de pruebas de la aplicación web.

Evidencias del desarrollo de la prueba:

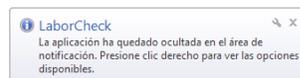
1.



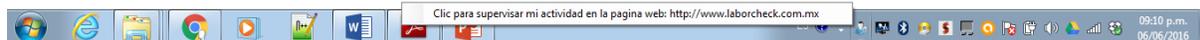
2.



3.



4.



Observaciones: La aplicación validó correctamente los datos del usuario que inicio sesión y posteriormente se minimizó en el área de notificación de la barra de tareas.

CP_AUC_3

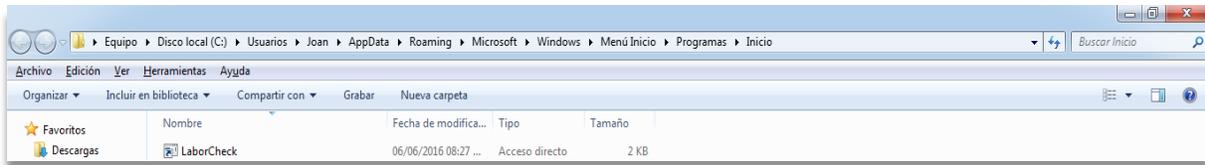
Aprobado

Desarrollo de la prueba:

1. Comprobar que la aplicación genere un acceso directo dentro del directorio que corresponde al grupo de programas de inicio del usuario, en Windows 7 se encuentra en la ubicación: C:\Users\Joan\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup.
2. Reiniciar el sistema operativo Windows. La aplicación se deberá reiniciar automáticamente después de iniciar sesión en el sistema operativo y procederá a minimizarse en la barra de tareas para continuar monitoreando las interacciones del teclado, mouse y ventanas.

Evidencias del desarrollo de la prueba:

1.



2.



Observaciones: El acceso directo se creó correctamente, lo cual permitió que la aplicación se reiniciara automáticamente después de reiniciar el sistema operativo Windows.

CP_AUC_7

Aprobado

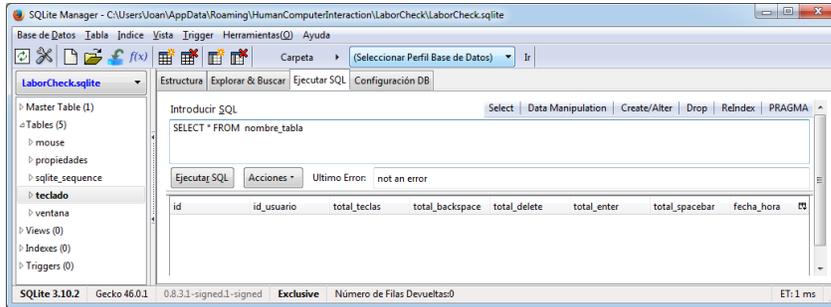
Desarrollo de la prueba:

1. Comprobar que la información de las variables de interacción usuario-computadora se guarde correctamente en la base de datos local. La base de datos local se ubica en la ruta de acceso de los datos de aplicación de un usuario de Windows, en Windows 7 se encuentra en: C:\Users\Joan\AppData\Roaming\LaborCheck\LaborCheck.sqlite, los registros almacenados deberán tener la fecha y hora de registro del lapso de tiempo de 5 minutos. Para poder abrir la base de datos se deberá usar alguna aplicación para gestionar la base de datos SQLite, por ejemplo, la aplicación del Navegador Firefox SQLite Manager.

Evidencias del desarrollo de la prueba:

1.

SQLite Manager



Interacciones con el teclado almacenadas cada cinco minutos

id	id_usuario	total_teclas	total_backspace	total_delete	total_enter	total_spacebar	fecha_hora
576	15	208	49	1	0	11	2016-06-01 20:21:25
577	15	74	4	0	3	0	2016-06-01 20:26:25
578	15	40	1	0	0	0	2016-06-01 20:31:25

Interacciones con el mouse almacenadas cada cinco minutos

id	id_usuario	total clics	total clics izq	total clics der	total_scroll_arr	total_scroll_ab	total_pixeles	fecha_hora
576	15	125	121	4	21	78	8458	2016-06-01 20:21:25
577	15	104	103	1	46	39	9375	2016-06-01 20:26:25
578	15	136	132	4	0	0	10347	2016-06-01 20:31:25

Interacciones entre ventanas almacenadas cada 5 minutos

id	id_usuario	total_ventanas_abiertas	total_cambios	fecha_hora
576	15	6	22	2016-06-01 20:21:25
577	15	5	13	2016-06-01 20:26:25
578	15	6	23	2016-06-01 20:31:25

Observaciones: Las interacciones usuario-computadora se almacenaron correctamente en la base de datos local, estas interacciones quedan temporalmente almacenadas mientras no se envíen al servidor remoto.

CP_AUC_8

Aprobado

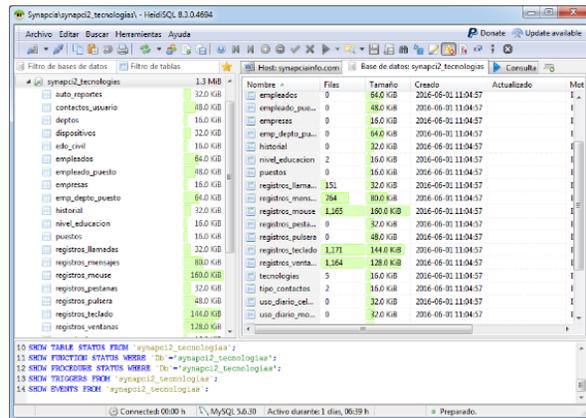
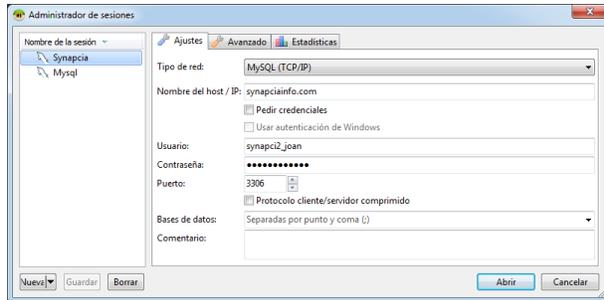
Desarrollo de la prueba:

1. Comprobar que los registros almacenados localmente son enviados y almacenados correctamente en la base de datos remota. Para tener acceso a la base de datos remota se deberán solicitar las credenciales para ingresar al servidor MySQL y capturarlas en algún software cliente como HeidiSQL.

Evidencias del desarrollo de la prueba:

1.

Software cliente - HeidiSQL



Interacciones del teclado almacenadas en la base de datos remota

7803	208	49	1	0	11	2016-06-01 20:21:25	15	1
7804	74	4	0	3	0	2016-06-01 20:26:25	15	1
7805	40	1	0	0	0	2016-06-01 20:31:25	15	1

Interacciones con el mouse almacenadas en la base de datos remota

9513	125	121	4	21	78	8458	2016-06-01 20:21:25	15	2
9514	104	103	1	46	39	9375	2016-06-01 20:26:25	15	2
9515	136	132	4	0	0	10347	2016-06-01 20:31:25	15	2

Interacciones entre ventanas almacenadas en la base de datos remota

11071	6	22	2016-06-01 20:21:25	15	3	40
11072	5	13	2016-06-01 20:26:25	15	3	40
11073	6	23	2016-06-01 20:31:25	15	3	40

Observaciones: Las interacciones usuario-computadora registradas en la base de datos local se recibieron y almacenaron correctamente en la base de datos remota.

Anexo E. Casos de prueba para la aplicación de interacción social

CP_AS_1

Aprobado

Desarrollo de la prueba:

1. Ejecutar la aplicación de interacción social que se encuentra dentro de la lista de aplicaciones instaladas. La aplicación deberá mostrar la pantalla de inicio de sesión.
2. Dar clic sobre el botón "login" de la pantalla de inicio dejando vacío alguno de los campos solicitados en el formulario. La aplicación deberá notificar que los campos deben capturarse.
3. Comprobar que no se permita iniciar la aplicación de monitoreo cuando no se tenga acceso a internet. La aplicación deberá mostrar un mensaje de esta validación. Esta validación es importante para asociar los datos con el usuario remoto.
4. Capturar sobre el formulario datos que no se encuentren almacenados en la base de datos remota. La aplicación deberá mostrar un mensaje de validación.
5. Capturar los datos correctos en formulario. Se consideran correctos cuando coinciden con los datos almacenados en la base de datos remota, es decir que estos datos debieron ser registrados anteriormente por medio de la aplicación de interacción usuario-computadora. La aplicación de interacción social deberá mostrar un mensaje notificando al usuario sobre la información que será recuperada.
6. La aplicación mostrará la pantalla de bienvenida con un menú en donde el usuario deberá administrar sus contactos. La pantalla también mostrará un hipervínculo hacia la aplicación web donde el usuario podrá consultar la información monitoreada.

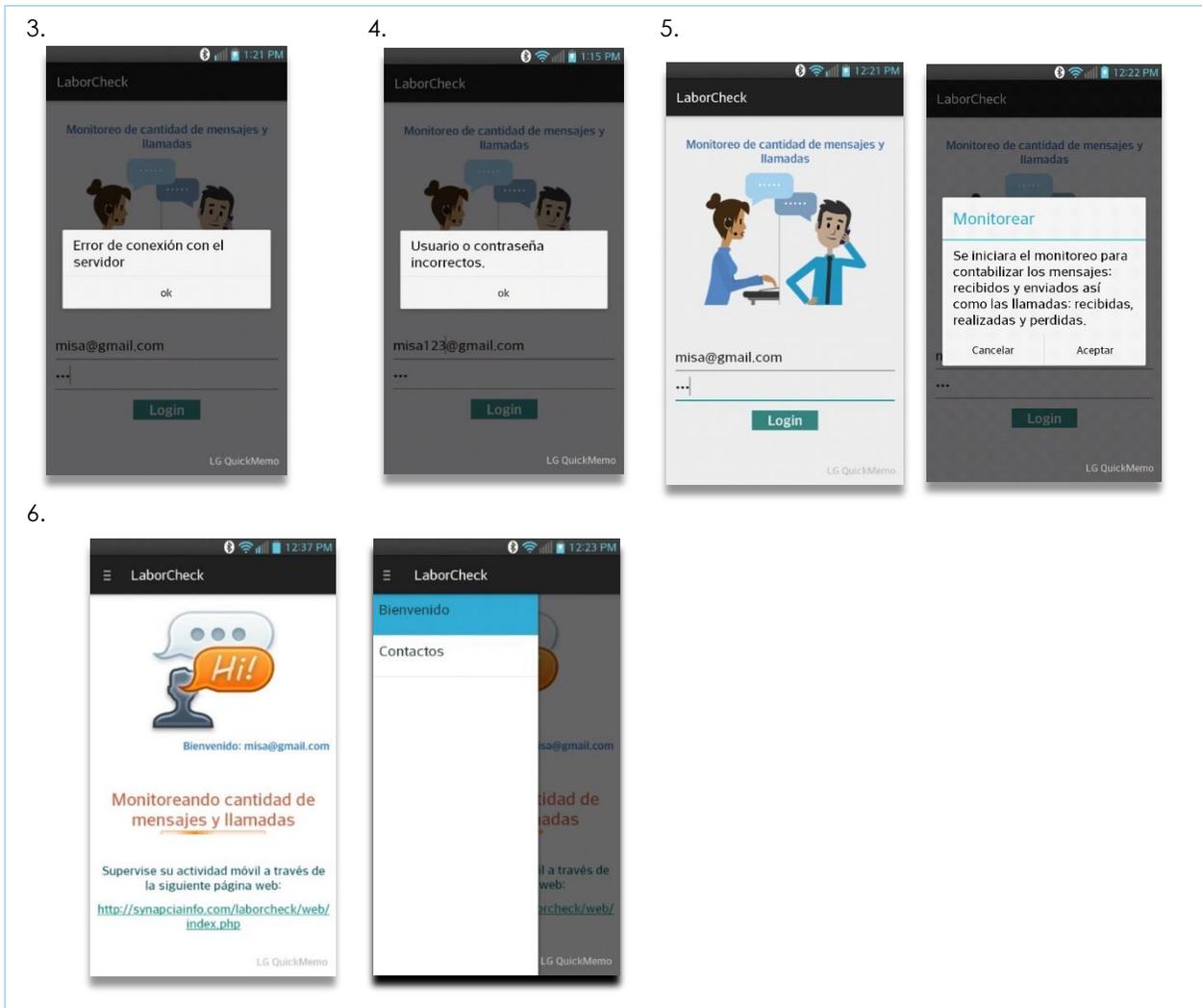
Evidencias del desarrollo de la prueba:

1.



2.





Observaciones: La aplicación inició correctamente con el usuario que se monitorea, incluso se giró horizontalmente la pantalla del celular y se comprobó que la aplicación se visualiza correctamente con las validaciones esperadas sin distorsionarse al mostrarse con la pantalla horizontal.

CP_AS_2

Aprobado

Desarrollo de la prueba:

1. Apagar o reiniciar el dispositivo móvil. Después de reiniciar el sistema la aplicación deberá permanecer con la misma cuenta de usuario que se está monitoreando. Se deberán reiniciar los servicios de la aplicación automáticamente:
 - PhoneService. Servicio para detección de mensajes y llamadas.
 - StoreRegistrosService. Servicio que inserta los mensajes y llamadas recolectados dentro de un lapso de dos horas en la base de datos local.
 - SendInformationService. Servicio que envía los registros locales hacia la base de datos remota cada dos horas y media.

Evidencias del desarrollo de la prueba:

1.



Observaciones: La aplicación se reinició automáticamente con el usuario autenticado inicialmente. Los servicios de monitoreo también se reiniciaron correctamente.

CP_AS_3

Aprobado

Desarrollo de la prueba:

1. Dar clic sobre la opción "contactos" en el menú de bienvenida. La aplicación mostrará un listado en donde se muestran los contactos agregados.
2. Dar clic en la opción "Agregar contacto". La aplicación mostrará un formulario con los campos requeridos del contacto como son: nombre, teléfono y tipo de contacto para clasificarlo como personal o laboral. En este formulario el usuario tendrá la opción de capturar los datos requeridos manualmente o seleccionarlos de la agenda del dispositivo móvil.
3. Dar clic en el botón "agregar" dejando los campos del formulario vacíos. El sistema deberá mostrar los mensajes de validación correspondientes.
4. Dar clic en la opción "Obtener de la agenda". La aplicación deberá obtener los contactos de la agenda contenida en el dispositivo móvil, los mostrará en un listado que incluye una opción de búsqueda.
5. Usar la opción de búsqueda para filtrar el contacto que se pretende agregar, una vez encontrado dar clic sobre el registro para mostrar los datos en el formulario que contiene los campos requeridos, dar clic sobre el botón "Agregar" y esperar a que el registro sea insertado en la lista de contactos. Finalmente, la aplicación notificara la inserción correcta con un mensaje mostrado en un cuadro de notificación.

Evidencias del desarrollo de la prueba:

1.



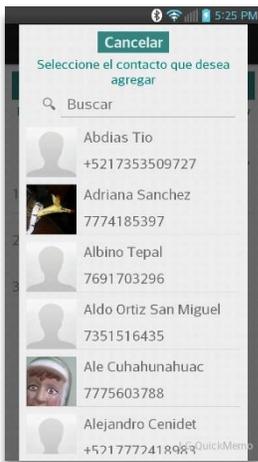
2.



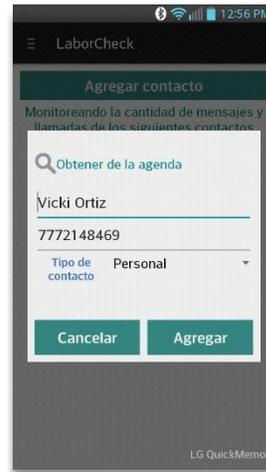
3.



4.



5.



Observaciones: El registro se insertó correctamente, se usaron los elementos disponibles para registrarlo de manera más rápida al tomar los datos de la agenda del dispositivo móvil.

CP_AS_4

Aprobado

Desarrollo de la prueba:

1. Dar clic sobre alguno de los registros que se muestran en el listado de contactos. La aplicación mostrará las opciones disponibles del contacto.
2. Dar clic sobre la opción "editar". La aplicación mostrará un formulario para poder modificar la información de los campos. Ingresar un número que ya exista en los registros. La aplicación mostrará un mensaje de validación indicando que no se permite modificar el registro con un número existente de otro contacto.
3. Capturar un número que no exista en la lista de contactos o dejar el mismo número telefónico y

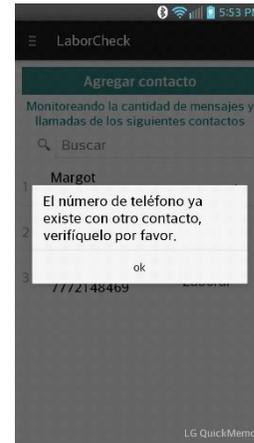
dar clic en guardar, los cambios se reflejarán inmediatamente sobre el listado de contactos notificando la actualización correcta con un mensaje mostrado en un cuadro de notificación.

Evidencias del desarrollo de la prueba:

1.



2.



3.



Observaciones: Los datos del contacto se actualizaron correctamente tomando en cuenta las validaciones de la aplicación.

CP_AS_5

Aprobado

Desarrollo de la prueba:

1. Dar clic sobre alguno de los contactos mostrados en la lista de los contactos del usuario. La aplicación mostrará las opciones disponibles del contacto.
2. Dar clic sobre la opción "Eliminar". La aplicación mostrará un mensaje donde el usuario deberá confirmar la eliminación del registro.

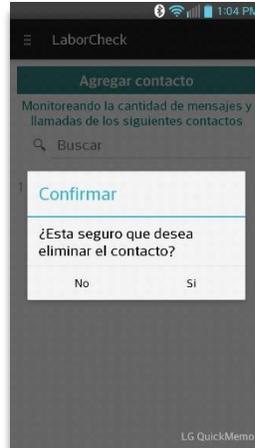
3. Dar clic en la opción "si" para confirmar la eliminación del registro. La aplicación eliminará el registro notificando la eliminación correcta con un mensaje mostrado en un cuadro de notificación y eliminará el contacto del listado que se muestra al usuario.

Evidencias del desarrollo de la prueba:

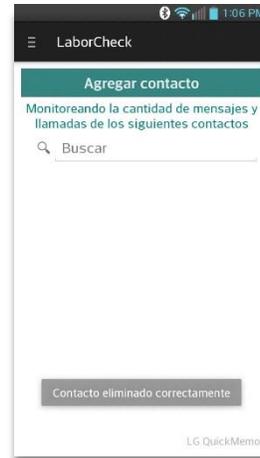
1.



2.



3.



Observaciones: La aplicación eliminó el registro correctamente y notificó al usuario con un mensaje de notificación mostrado en la parte inferior de la pantalla.

CP_AS_8

Aprobado

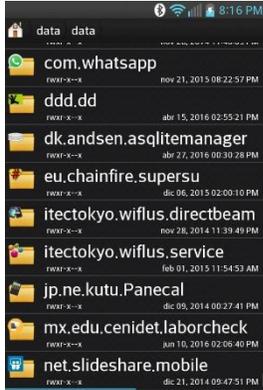
Desarrollo de la prueba:

1. Comprobar que el servicio que almacena los registros de mensajes y llamadas localmente este en ejecución. El nombre del servicio es "StroreRegistrosService", el cual se puede consultar desde las aplicaciones activas del dispositivo móvil. Para comprobar que la información este almacenada en la base de datos local se debe configurar el dispositivo móvil en modo "root" para tener acceso a esta información debido a que el sistema operativo Android bloquea esta información por motivos de seguridad. Comprobar que la cantidad de mensajes y llamadas se registren correctamente. Los registros almacenados se registran tomando la información de las tablas que contabilizan los mensajes y llamadas. Esta información se recupera cada dos horas.

Evidencias del desarrollo de la prueba:

1.

Carpeta “mx.edu.cenidet.laborcheck” que almacenada la base de datos local.



Cantidad de mensajes registrados en la tabla “registro_mensajes”:

id	id_usuario	total_rec_lab	total_rec_per	total_env_lab	total_env_per	fecha_hora
1	15	1	1	3	2	04-06-2016 21:15:13

Cantidad de llamadas registradas en la tabla “registro_llamadas”:

id	id_usuario	total_sal_lab	total_sal_per	total_ent_lab	total_ent_per	total_perd_lab	total_perd_per	fecha_hora
1	15	3	2	1	1	1	1	04-06-2016 21:15:13

Observaciones: La cantidad de mensajes y llamadas se almacenó correctamente con un lapso de tiempo de dos horas.

CP_AS_9

Aprobado

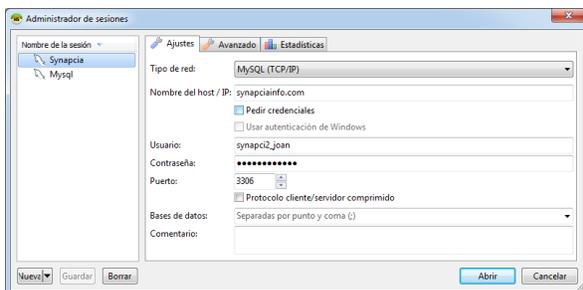
Desarrollo de la prueba:

1. Comprobar que el servicio “SendInformationService”, el cual envía los registros locales a la base de datos remota se esté ejecutando dentro de las aplicaciones activas del dispositivo móvil. Disponer de un usuario y una contraseña de acceso hacia la base de datos remota para configurar algún software que actúe como cliente y permita gestionar la base de datos, una vez configurado, generar una consulta sobre las tablas de “registros_mensajes” y “registros_llamadas” con el fin de comprobar que los registros locales se almacenen correctamente.

Evidencias del desarrollo de la prueba:

1.

HeidiSQL. Software cliente para gestionar la base de datos remota



Registros de mensajes:

🔑 id_registro_mensaje	msj_ent_trab	msj_sal_trab	msj_ent_pers	msj_sal_pers	fecha_hora
1535	0	0	0	0	2016-06-04 19:30:10
1536	0	0	0	0	2016-06-04 19:49:46
1537	1	3	1	2	2016-06-04 21:15:13
1538	0	0	0	0	2016-06-04 20:53:00
1539	0	0	0	0	2016-06-04 21:20:01

Registros de llamadas:

🔑 id_registro_llamada	llam_ent_trab	llam_sal_trab	llam_per_trab	llam_ent_pers	llam_sal_pers	llam_per_pers	fecha_hora
1334	0	0	0	0	0	0	2016-06-04 19:10:01
1335	0	0	0	0	1	0	2016-06-04 19:30:10
1336	0	0	0	0	0	0	2016-06-04 19:49:46
1337	1	3	1	1	2	1	2016-06-04 21:15:13
1338	0	0	0	0	0	0	2016-06-04 20:53:36

Observaciones: Los mensajes y llamadas almacenados localmente se recibieron correctamente en la base de datos remota.

Anexo F. Casos de prueba para la aplicación web

CP_AW_1

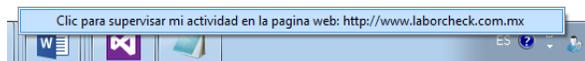
Aprobado

Desarrollo de la prueba:

1. Dar clic sobre el enlace incluido en la aplicación de interacción usuario-computadora o dar clic sobre el hipervínculo incluido en la aplicación de interacción social.
2. Dar clic en el botón de "Login" con los campos vacíos. La aplicación deberá mostrar los mensajes de validación y denegar el acceso a la aplicación web.
3. Capturar algún dato incorrecto ya sea correo o contraseña. La aplicación deberá mostrar el mensaje de validación correspondiente y negar el acceso.
4. Capturar el nombre de usuario y contraseña correspondientes al usuario que desea consultar su actividad computacional y móvil. La aplicación mostrará el módulo inicial de consulta.

Evidencias del desarrollo de la prueba:

1.



2.



3.



4.



Observaciones: La aplicación web validó correctamente los datos de acceso y mostró la pantalla de consulta al usuario autenticado correctamente.

cenidet[®]
*Centro Nacional de Investigación
y Desarrollo Tecnológico*