



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

INFORME DE:



Pasantía



Ayudante de Cátedra



Práctica Pre Profesional No Remunerada



Ayudante de Investigación



Servicio a la comunidad

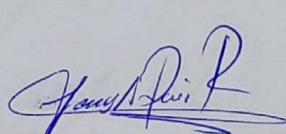
COMANDO CONJUNTO FF. AA

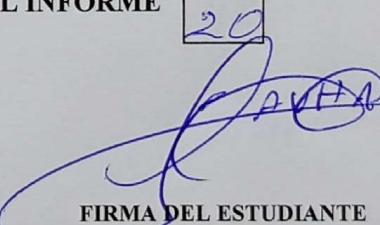
NOMBRES Y APELLIDOS DEL ESTUDIANTE: ALAN VLADIMIR HERRERA ARAUJO

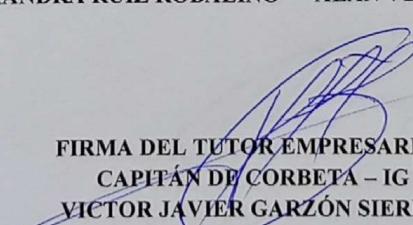
NOMBRES Y APELLIDOS DEL TUTOR ACADÉMICO: JENNY ALEXANDRA RUIZ ROBALINO

CALIFICACIÓN DEL INFORME

20


FIRMA DE TUTOR(A) ACADÉMICO(A)
ING. JENNY ALEXANDRA RUIZ ROBALINO


FIRMA DEL ESTUDIANTE
ALAN VLADIMIR HERRERA ARAUJO


FIRMA DEL TUTOR EMPRESARIAL
CAPITÁN DE CORBETA - IG
VICTOR JAVIER GARZÓN SIERRA



Sangolquí, 15/02/2025

1. INTRODUCCIÓN

El presente informe describe las actividades realizadas durante las prácticas preprofesionales en el Comando Conjunto de las Fuerzas Armadas (COMACO), institución encargada de la seguridad y defensa del territorio. Estas prácticas, desarrolladas del 24 de septiembre al 26 de noviembre de 2024 como parte de la formación en Tecnologías de la Información, permitieron aplicar conocimientos en desarrollo de software y administración de redes.

En el área de desarrollo de software, las actividades se enfocaron en la revisión y actualización del código fuente del aplicativo móvil de la institución. Se implementaron mejoras en el rendimiento y seguridad del sistema, optimizando el proceso de autenticación y carga de datos. Además, se incorporó un sistema de alertas en la sección de visualización de informes, así como una funcionalidad para notificar a los usuarios sobre la conectividad del aplicativo con el servidor, permitiendo identificar de manera oportuna problemas de conexión. Adicionalmente, se documentaron las modificaciones realizadas para facilitar futuras actualizaciones y reducir la dependencia de terceros.

En el ámbito de redes, se colaboró en la configuración y optimización de dispositivos de red para mejorar la estabilidad y eficiencia de la conectividad dentro del departamento asignado. Se identificaron y solucionaron problemas de inestabilidad, optimizando la infraestructura para garantizar una comunicación más fluida entre los sistemas y usuarios.

Los resultados obtenidos reflejan una mejora significativa en la operatividad de los sistemas tecnológicos del COMACO, optimizando tanto el desempeño del aplicativo móvil como la estabilidad de la red. Asimismo, la experiencia permitió consolidar habilidades técnicas y prácticas en la gestión de infraestructuras tecnológicas, contribuyendo al desarrollo profesional del practicante y al fortalecimiento de los procesos tecnológicos de la institución.

2. PROGRAMA DE PRACTICAS PREPROFESIONALES

FASE	FECHA INICIO	FECHA FIN	ACTIVIDAD	TAREAS	HORAS EXACTAS	HORAS	TOTAL	
Capacitación	24/9/2024	27/9/2024	Introducción a las herramientas.	Explorar las herramientas y conocer sus funciones principales.	3	6	24	
				Configurar las herramientas en el entorno de trabajo.	3			
			Capacitación en tecnologías.	Asistir a una sesión teórica sobre tecnologías clave.	3	6		
				Realizar ejercicios prácticos para aplicar los conceptos aprendidos.	3			
			Taller práctico sobre el uso de herramientas.	Crear un proyecto básico usando las herramientas aprendidas.	6	6		
			Evaluación de conocimientos	Realizar un examen práctico para evaluar lo aprendido y Revisión los resultados y recibir retroalimentación.	6	6		

FASE	FECHA INICIO	FECHA FIN	ACTIVIDAD	TAREAS	HORAS EXACTAS	HORAS	TOTAL
			Inicio del proyecto con una revisión exhaustiva del código para identificar posibles errores generales y áreas problemáticas.	Revisión del código para identificar errores generales y áreas problemáticas Documentación de hallazgos Iniciales en base a la revisión de código.	4 2	6	
			Identificación de secciones obsoletas en las librerías que podrían requerir actualización o eliminación.	Analisis de las librerías obsoletas y su funcionalidad actual Listar posibles alternativas y priorizar áreas de atención	3 3	6	
			Búsqueda inicial de librerías actualizadas y complementarias para reemplazar o mejorar las secciones obsoletas (Parte 1).	Realizar una investigación de las librerías y sus actualizaciones para identificar aquellas que pueden reemplazar o mejorar las secciones obsoletas en el código. Evaluar las nuevas funcionalidades y mejoras ofrecidas por estas librerías en relación con las necesidades del proyecto.	4 2	6	
			Continuación de la búsqueda de librerías para cubrir todas las necesidades identificadas en el análisis anterior (Parte 2).	Analisis Detalladas de las Dependencias de las Librerías Seleccionadas Descarga y configuración inicial de las nuevas librerías	3 3	6	
			Analisis de los errores encontrados y evaluación de posibles soluciones.	Realizar un análisis exhaustivo de los errores encontrados durante las pruebas iniciales del sistema, identificando los tipos de errores (funcionales, de rendimiento, de compatibilidad, etc.). Clasificar los errores según su gravedad y su impacto en la funcionalidad de la aplicación para priorizar su resolución.	3 3	6	
			Implementación de las correcciones basadas en los errores detectados y en las librerías seleccionadas.	Realizar un análisis detallado de los resultados obtenidos tras la implementación de las correcciones en el código sobre sus librerías Ajustes en el código y librerías según los errores identificados	4 2	6	
			Primeras rondas de pruebas para evaluar la efectividad de las correcciones aplicadas y análisis de los resultados obtenidos (Parte 1).	Realizar la ejecución de pruebas para verificar si las correcciones aplicadas en el código y las actualizaciones de librerías han resuelto los problemas detectados en pruebas anteriores. Evaluar el comportamiento del sistema, asegurando que las correcciones no introduzcan nuevos problemas.	4 2	6	
			Aplicación de correcciones adicionales utilizando nuevas versiones de las librerías identificadas en las anteriores.	Aplicación de nuevas versiones de librerías para resolver errores restantes Asegurarse de que las nuevas versiones de las librerías sean compatibles con el sistema y no introduzcan conflictos o nuevos problemas.	3 3	6	
			Pruebas y análisis detallados de los cambios realizados, enfocándose en la resolución de los errores restantes (Parte 2).	Realizar la ejecución para verificar las actualizaciones de librerías han sido realizado.	6	6	
			Revisión exhaustiva del código modificado para identificar posibles errores detectados en las rondas de pruebas anteriores.	Realizar una revisión detallada del código modificado tras las correcciones y actualizaciones realizadas en las rondas anteriores, buscando posibles errores o inconsistencias que puedan haber surgido.	6	6	
			Descarga, integración y configuración de nuevas librerías y componentes para mejorar el rendimiento del sistema.	Descarga e integración de nuevas librerías y componentes Configuración y ajuste inicial de las librerías integradas	4 2	6	
			Pruebas y corrección de problemas relacionados con la integración de las nuevas librerías y componentes.	Realizar pruebas iniciales para verificar la correcta integración de las nuevas librerías y componentes en el sistema. Identificar y corregir los problemas detectados durante las pruebas de integración, tales como incompatibilidades, fallos en la funcionalidad o errores en la comunicación entre componentes.	4 2	6	
			Implementación de nuevas librerías durante las pruebas.	Implementar nuevas librerías en el proyecto durante las fases de prueba para asegurar que todas las funcionalidades necesarias sean cubiertas. Asegurar que las librerías implementadas no generen conflictos con el código existente y que cumplan con los requisitos del proyecto.	4 2	6	
			Pruebas de las nuevas librerías implementadas desde un IDE (Parte 1).	Identificar y registrar cualquier problema o error que surja durante las pruebas realizadas en el IDE..	6	6	
			Identificación y corrección de errores detectados durante las pruebas en el IDE, con ajustes necesarios.	Ánalisis de errores detectados durante las pruebas en el IDE Ajustes y validaciones para corregir los errores identificados	3 3	6	
			Segunda ronda de pruebas de las mejoras implementadas desde un IDE.	Evaluar si los problemas detectados en la primera ronda de pruebas han sido resueltos y si está operando correctamente y analizar el rendimiento del sistema tras las mejoras y asegurarse de que no se hayan introducido nuevos errores o conflictos.	6	6	
			Elaboración del borrador del documento de actualización de la aplicación, detallando los cambios realizados y los resultados obtenidos.	Redactar el borrador del documento de actualización de la aplicación, describiendo detalladamente todos los cambios realizados durante el proceso de desarrollo, como actualizaciones de librerías y correcciones de errores. Revisión del documento para asegurar claridad y precisión	4 2	6	

FASE	FECHA INICIO	FECHA FIN	ACTIVIDAD	TAREAS	HORAS EXACTAS	HORAS	TOTAL
Configuración inicial de la app móvil	25/10/2024	29/10/2024	<p>Revisión detallada de los servicios (Base de datos y autenticación móvil) en la app, identificando y analizando en qué partes del código estaban siendo llamados.</p> <p>Modificación y pruebas del código para validar correctamente los llamados a los servicios y asegurar su correcta integración.</p> <p>Pruebas para garantizar que los servicios están siendo llamados de manera adecuada y eficiente, ajustando cualquier error encontrado.</p>	<p>Inspeccionar el código fuente de la aplicación para identificar en qué partes del código se están realizando los llamados a los servicios de base de datos.</p> <p>Revisar el código relacionado con los métodos de autenticación móvil, asegurándose de que se implementen adecuadamente las técnicas de seguridad y que el flujo de autenticación funcione correctamente.</p> <p>Continuar a documentación anterior y agregar los resultados encontrados durante la inspección de los servicios de base de datos y autenticación móvil, detallando cualquier problema o área de mejora.</p> <p>Modificar el código para corregir cualquier error en los llamados a los servicios, asegurándose de que se realicen de manera eficiente y segura.</p> <p>Validar que los servicios se estén llamando correctamente y que las respuestas sean las esperadas.</p> <p>Revisar los logs generados durante las pruebas para detectar cualquier error o inconsistencia en los llamados a los servicios.</p> <p>Evaluar el tiempo de respuesta, la eficiencia en el uso de recursos y la estabilidad de los servicios durante las pruebas.</p> <p>Realizar ajustes menores en el código para corregir cualquier error o inconsistencias encontradas durante las pruebas.</p> <p>Comprobar que todos los servicios funcionen de manera eficiente y efectiva, resolviendo cualquier posible problema antes de la implementación en producción.</p>	2 2 2 2 2 2 2 2 2 2	6 6 6	18

FASE	FECHA INICIO	FECHA FIN	ACTIVIDAD	TAREAS	HORAS EXACTAS	HORAS	TOTAL	
Mejoras en la app móvil	31/10/2024	8/11/2024	Levantamiento de requerimientos	Identificar áreas críticas que necesiten mejoras o actualizaciones en el sistema de autenticación.	1	6	24	
				Priorizar las funcionalidades según la importancia para el usuario final y el impacto en el sistema.	1			
				Realizar un análisis del rendimiento actual en la obtención de datos, evaluando tiempos de carga, procesos lentos y posibles cuellos de botella.	1			
				Evaluuar la viabilidad de cada mejora y su impacto en el rendimiento general.	1			
				Establecer qué tipos de eventos deben generar alertas inmediatas y cuáles pueden ser menos urgentes.	1			
			Implementación de mejoras en el login, optimizando la seguridad y la experiencia del usuario.	Definir el formato de las alertas, como mensajes dentro de la app, según la gravedad y urgencia de los eventos.	1	6		
				Modificación del sistema de autenticación para agregar validaciones adicionales implementando nuevos métodos de seguridad.	2			
			Desarrollo de alertas relacionadas con el ingreso de resultados en operaciones.	Validar que las nuevas validaciones y métodos de seguridad no interfieran con la experiencia de usuario y que el proceso de inicio de sesión funcione sin problemas.	1	6		
				Configurar un sistema de notificaciones para alertar a los usuarios sobre el ingreso de resultados en las operaciones.	2			
				Validar que las alertas lleguen a los usuarios de manera oportuna y sin fallos, y que la información sea precisa y clara.	1			
			Análisis del rendimiento de la carga de datos y detección de áreas críticas.	Identificar posibles cuellos de botella que puedan estar afectando la experiencia del usuario, como tiempos de carga excesivos o retrasos en la actualización de datos.	2	6		
				Comparar los tiempos de carga actuales con las métricas de rendimiento esperadas o estándar para identificar desviaciones y áreas que requieren optimización.	1			
				Mejorar la estructura y el acceso a los datos para reducir los tiempos de respuesta y acelerar la carga en la app.	2			
			Optimización de los procesos de carga de datos, mejorando la velocidad y eficiencia.	Comparar los resultados obtenidos con los tiempos de respuesta previos para asegurar que las optimizaciones hayan mejorado la eficiencia sin afectar otras funcionalidades de la app.	1	6		
				Realizar pruebas integrales para verificar que las mejoras en el login, alertas y carga de datos sigan funcionando correctamente tras las modificaciones adicionales, asegurando que no haya nuevos errores y que se mantenga o mejore el rendimiento.	3			
			Ejecución de pruebas sobre las mejoras en el login, alertas y carga de datos tras modificaciones extras.	Registro de los errores detectados y los pasos para solucionarlos.	2	6		
				Actualizar la documentación del proyecto, incluyendo las modificaciones realizadas en las mejoras del login, alertas y carga de datos.	1			

FASE	FECHA INICIO	FECHA FIN	ACTIVIDAD	TAREAS	HORAS EXACTAS	HORAS	TOTAL		
Pruebas de funcionalidad y ajustes en la app móvil	11/11/2024	19/11/2024	Identificar problemas comunes reportados dado la actualización en Android Studio.	Investigar si la actualización ha causado conflictos con versiones anteriores de librerías, dependencias o configuraciones específicas..	2	6	42		
			Identificación de ajustes necesarios y planeación de las modificaciones.	Identificación de las partes de la app que requieren modificaciones inmediatas	2				
				Establecer prioridades para las modificaciones, enfocándose primero en las áreas críticas que puedan afectar el rendimiento y la estabilidad de la app.	2				
				Identificar secciones del código que no aportan valor a la funcionalidad general de la app y que podrían eliminarse o simplificarse para mejorar el rendimiento y la legibilidad.	2				
			Análisis del código fuente para identificar redundancias y áreas de mejora.	Evaluar las posibles dificultades que puedan surgir al agregar nuevas funcionalidades o realizar modificaciones importantes debido a la forma en que está organizado el código.	1				
				Configuración de todas las librerías necesarias en Android Studio.	1				
				Verificación de las versiones necesarias del SDK para asegurar la compatibilidad.	1				
			Configuración inicial para la integración en Android Studio.	Configuración del proyecto dentro de Android Studio para integrarlo correctamente.	1				
				Validación de las configuraciones iniciales del proyecto.	2				
				Modificación de los parámetros del proyecto para mejorar el rendimiento o la compatibilidad.					
			Ajustes en la configuración de la app en Android Studio.	Identificación y resolución de posibles conflictos de configuraciones previas.					
				Implementación del código en Android Studio.	6	6			
				Implementar las correcciones y modificaciones en Android Studio, asegurándose de seguir las mejores prácticas para evitar errores adicionales.					
			Agregado de librerías faltantes para completar dependencias por Android Studio.	Revisión de las dependencias y asegurarse de que todas las librerías necesarias estén integradas.			1		
				Validación de que las nuevas librerías no generen conflictos con las versiones anteriores.			1		
				Corrección de errores detectados en las dependencias agregadas.			3		
			Pruebas preliminares de construcción del APK.	Identificación de problemas causados por las nuevas librerías agregadas.	2	6	6		
				Compilación preliminar del APK para identificar problemas.	2				
				Verificación de que el APK se pueda instalar correctamente en dispositivos Android.	1				
			Verificación final del código para evitar conflictos en la generación del APK.	Asegurarse de que no haya errores o conflictos en el código que puedan impedir la correcta compilación.	3	6	6		
				Validación	2				
				Generar el APK listo para su distribución o prueba.	1				
			Generación y obtención del APK.	Extraer el archivo generado para realizar pruebas más profundas en dispositivos reales.	1				
				Instalar y ejecutar el APK en varios dispositivos con diferentes tamaños y resoluciones de pantalla.	4	6	6		
				Detectar problemas de compatibilidad en dispositivos concretos, como problemas de visualización o funcionalidad.	2				
			Pruebas del APK en dispositivos de diferentes resoluciones.	Realizar pruebas del APK en una variedad mayor de dispositivos, para garantizar la compatibilidad en diferentes modelos.	4	6	6		
				Confirmar que la app sea completamente compatible con las versiones de los sistemas operativos de los dispositivos de prueba.	2				
				Validación final de la funcionalidad y compatibilidad de la app.					

FASE	FECHA INICIO	FECHA FIN	ACTIVIDAD	TAREAS	HORAS EXACTAS	HORAS	TOTAL	
Documentación de la app móvil	20/11/2024	20/11/2024	Revisión general del proyecto para identificar los aspectos más relevantes a incluir en la documentación.	Realización de un análisis exhaustivo de todos los aspectos del proyecto, incluidos los procesos de desarrollo, optimizaciones y cambios implementados.	2	6	30	
				Determinación de los elementos más importantes que deben ser incluidos en la documentación,	1			
			Redacción y finalización sobre la creación del Informe sobre el la actualización y mejora de la app.	Escribir un informe dando la parte introductoria que describa el objetivo principal del proyecto, su propósito y la importancia de las decisiones tomadas durante el desarrollo.	2	6		
				Concluir las conclusiones del informe, resumiendo los logros alcanzados, los desafíos enfrentados y las soluciones implementadas.	1			
Optimización de la infraestructura de redes	21/11/2024	22/11/2024	Revisión de la infraestructura de redes actual, identificación de equipos de red y protocolos utilizados.	Inspección detallada de los equipos de red existentes, como routers, switches y servidores.	2	6	30	
				Evaluación de los dispositivos y protocolos de comunicación empleados en la infraestructura, verificando su compatibilidad y capacidad.	1			
			Análisis de rendimiento de la red mediante herramientas de monitoreo.	Implementación de herramientas de monitoreo para evaluar el rendimiento de la red.	2	6		
				Análisis de los datos recolectados para identificar posibles problemas de rendimiento o de configuración en la red.	1			
Configuración del router	25/11/2024	25/11/2024	Implementación de mejoras en la infraestructura de redes.	Realización de cambios necesarios en la configuración de equipos de red y ajustes de protocolos para mejorar el rendimiento y la seguridad.	6	6	30	
				Inspección de los parámetros de configuración existentes, incluyendo seguridad, rendimiento y acceso.	3			
			Configuración avanzada del router para optimizar la seguridad y el rendimiento.	Verificación de los protocolos de enrutamiento y de comunicación, asegurando que estén configurados correctamente para evitar conflictos o pérdida de datos.	3	6		
				Realizar una revisión exhaustiva del documento, asegurándose de que todas las secciones estén completas y bien estructuradas.	3			
Presentación de la documentación del aplicativo móvil.	26/11/2024	26/11/2024	Entrega del informe final a los responsables del seguimiento del proyecto sobre el aplicativo móvil.	Entregar el informe final a los responsables del seguimiento del proyecto, asegurando que comprendan el contenido y las actualizaciones realizadas en el aplicativo móvil.	1	6	30	
				Aplicar los ajustes necesarios en la documentación, asegurándose de que todos los comentarios y sugerencias sean abordados de manera adecuada y entrega a los responsables del proyecto.	2			

3. DESARROLLO

En esta parte del desarrollo detallaré las actividades realizadas durante mis prácticas preprofesionales en el Comando Conjunto de las Fuerzas Armadas (COMACO), enfocándome en los distintos días de trabajo a lo largo del período de prácticas. El desarrollo estará organizado por fases, y cada fase corresponderá a un día de trabajo específico. En cada una de ellas, explicaré las actividades que se realizaron, los objetivos alcanzados y los conocimientos adquiridos.

Fase 1: Capacitación

Durante esta fase, se llevó a cabo un proceso de formación destinado a familiarizarse con el entorno de trabajo, las herramientas tecnológicas y la estructura de los sistemas utilizados en el COMACO. Inicialmente, se realizó una introducción general sobre la infraestructura técnica de la aplicación móvil y la página web, abordando sus

principales componentes y la interrelación entre ellos. Posteriormente, se profundizó en las tecnologías empleadas en el desarrollo del aplicativo móvil, detallando su arquitectura, el uso de bases de datos y las herramientas de programación utilizadas. Como parte de la capacitación, se llevó a cabo un taller práctico en el que se aplicaron los conocimientos adquiridos, permitiendo la interacción directa con el entorno de desarrollo, la edición de código y la implementación de nuevas funciones. Finalmente, se realizó una evaluación teórico-práctica para medir la comprensión de los conceptos abordados, identificando fortalezas y áreas de mejora. Esta fase fue fundamental para adquirir el conocimiento necesario que permitió abordar de manera efectiva las actividades de desarrollo y optimización en las siguientes etapas de las prácticas.

Fase 2: Revisión del Código de la App Móvil.

Durante esta fase, se llevó a cabo una revisión exhaustiva del código de la aplicación móvil con el objetivo de mejorar su calidad y asegurar su eficiencia. El proceso comenzó con una revisión general del código utilizando Visual Studio Code, identificando posibles errores y áreas de mejora. No se utilizó GitHub en esta etapa debido a que el código fue entregado de manera local. A lo largo de esta fase, se adoptó una metodología ágil adaptada, lo que permitió trabajar en tareas bien definidas y realizar mejoras sin interrupciones.

Se identificaron secciones de código obsoletas y se actualizó el código aplicando principios de "código limpio". Se llevó a cabo una búsqueda de nuevas librerías para reemplazar las secciones obsoletas, mejorando la estabilidad y el rendimiento de la aplicación. A medida que se avanzaba, se implementaron correcciones y se realizaron pruebas para verificar la efectividad de los cambios. A lo largo de las semanas, se llevaron a cabo múltiples rondas de pruebas y ajustes, mejorando tanto el rendimiento de la app como su estabilidad con las diferentes librerías aplicadas.

El proceso de pruebas también se extendió a la verificación de la integración de nuevas librerías y la resolución de problemas relacionados con la compatibilidad de versiones. Se implementaron nuevas librerías que mejoraron el rendimiento del sistema, y se realizaron pruebas continuas para asegurar que la aplicación funcionara correctamente en distintos entornos y condiciones de red. Los resultados de las pruebas fueron positivos, y las correcciones permitieron optimizar la aplicación, corrigiendo errores y mejorando la experiencia del usuario.

Finalmente, se elaboró un documento detallado que registró todos los cambios realizados en la fase de revisión del código. Este documento incluye los errores corregidos, las librerías actualizadas, y las mejoras en el rendimiento y la estabilidad de la aplicación, sirviendo como referencia para futuros desarrollos y actualizaciones.

Imágenes relacionadas:

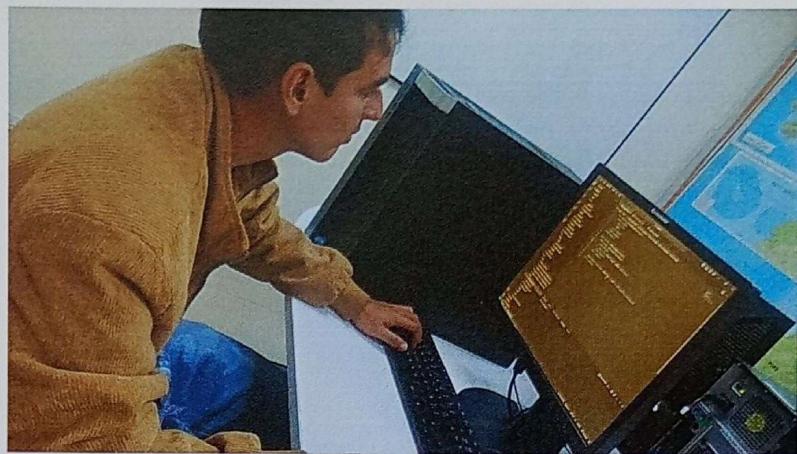


Figura 1: Fotografía durante la revisión inicial del código usando el IDE Visual Studio Code.

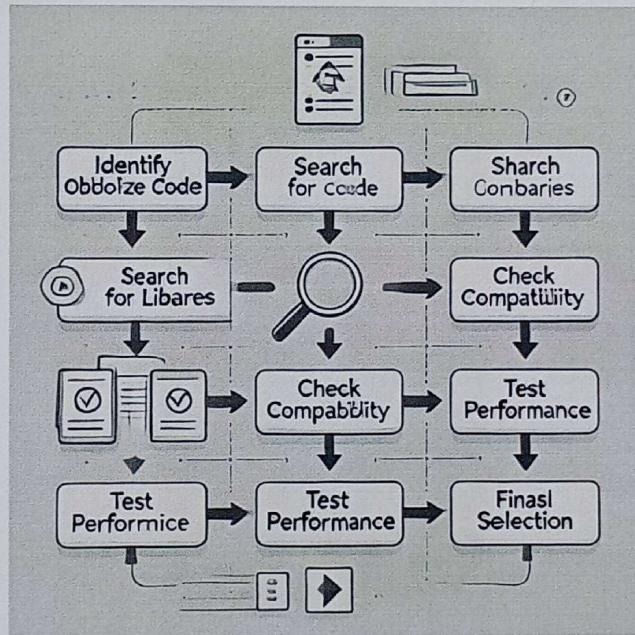


Figura 2: Diagrama de flujo que ilustra el proceso de selección de librerías.



Figura 3: Fotografía referente a la revisión del código.

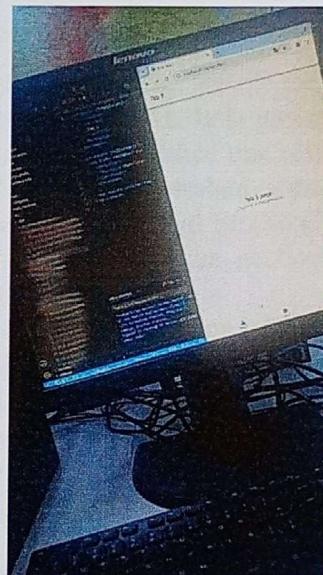


Figura 4: Fotografía del proceso de depuración en Visual Studio Code.



Figura 5: Fotografía mostrando los resultados de las pruebas unitarias en Visual Studio Code.

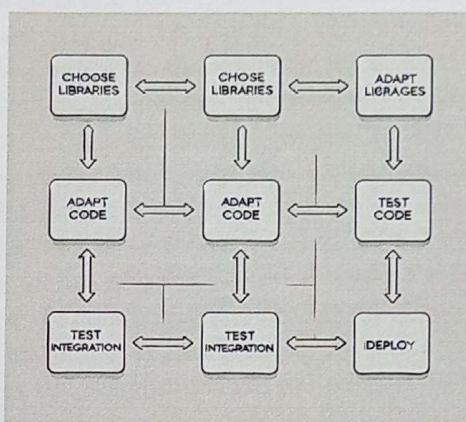


Figura 6: Diagrama de flujo con el proceso de integración de nuevas librerías en el código.

Fase 3: Configuración inicial de la app móvil.

En la fase de configuración inicial de la app móvil, me enfoqué en revisar y asegurar la correcta integración de los servicios clave de la aplicación, como la base de datos y la autenticación móvil. El 25 de octubre, realicé una revisión detallada del código para identificar en qué partes del mismo se estaban llamando estos servicios y cómo interactuaban con la aplicación. Utilicé Visual Studio Code para analizar la estructura del código, centrándome en las secciones que gestionaban la comunicación con el backend y el sistema de autenticación.

Durante esta revisión, detecté que algunos de los servicios no estaban siendo llamados correctamente o presentaban problemas de integración debido a las recientes actualizaciones de librerías. Este tipo de inconsistencias afectaba el rendimiento y la fiabilidad de la app, por lo que el siguiente paso fue modificar y ajustar el código. El 28 de octubre, me dediqué a corregir las rutas de los llamados a los servicios y a actualizar las librerías asociadas, con el objetivo de optimizar la comunicación entre la app y los servicios backend.

Tras realizar las modificaciones, utilicé Visual Studio Code para realizar pruebas, verificando que los cambios realizados no generaran nuevos errores y que los servicios fueran llamados de manera eficiente. Sin embargo, algunos problemas persistieron, especialmente relacionados con la velocidad de respuesta y la integración de los servicios, lo que requería más ajustes para garantizar su correcto funcionamiento.

El 29 de octubre, realicé pruebas adicionales, ahora con un enfoque en la optimización de la velocidad de respuesta de los servicios, en especial de la autenticación. Identifiqué y corregí errores menores que impactaban la rapidez de las respuestas. También realicé pruebas de integración, verificando cómo los diferentes servicios interactuaban entre sí, asegurándome de que la app pudiera manejar múltiples solicitudes sin comprometer el rendimiento.

Como resultado de esta fase, se logró mejorar la conexión de los servicios y optimizar su rendimiento, solucionando muchos de los problemas causados por la incompatibilidad de las librerías actualizadas, lo que permitió avanzar en la fase de configuración de la app.

Fase 4: Mejoras en la app móvil

Durante la fase de mejoras en la app móvil, se abordaron varios aspectos clave para optimizar la seguridad, la experiencia del usuario y el rendimiento de la aplicación. El proceso comenzó con el levantamiento de requerimientos, en el cual se identificaron áreas críticas que necesitaban optimización. Se centró principalmente en la mejora del sistema de autenticación, la entrada de datos en las operaciones y la conectividad con los servicios backend y la base de datos. A partir de este análisis, se detectaron problemas como la inestabilidad en la conectividad con los servicios y errores en la entrada de datos, lo que afectaba la experiencia del usuario. Se establecieron criterios para la implementación de alertas que notificaran al usuario sobre eventos importantes, como caídas en la conectividad o errores en el ingreso de datos.

Con estos requerimientos en mente, se implementaron varias mejoras significativas. Una de las más importantes fue la optimización del sistema de login. Se utilizó CryptoJS para cifrar datos sensibles y JWT (JSON Web Tokens) para gestionar de manera segura las sesiones de usuario, asegurando que solo los usuarios autorizados pudieran acceder a la app. La implementación de estos mecanismos de seguridad fortaleció la protección contra accesos no autorizados. Además, se mejoró la experiencia del usuario al hacer que el proceso de autenticación fuera más claro y eficiente. Aunque las pruebas iniciales en Visual Studio Code fueron exitosas, se encontraron problemas cuando se trasladaron a Android Studio. La app tenía dificultades para realizar los llamados al backend correctamente, lo que requirió ajustes adicionales en los servicios de autenticación para garantizar que las solicitudes se manejaran de manera correcta. Tras realizar estos ajustes, se consiguió una autenticación más estable, con tiempos de respuesta optimizados.

Otra mejora importante fue el desarrollo de alertas. Se implementaron alertas que se activan cuando los usuarios ingresan resultados en las operaciones. Estas alertas proporcionan retroalimentación inmediata, informando al usuario si los datos fueron registrados correctamente o si se produjo un error. Además, se desarrollaron alertas específicas para monitorear la conectividad de la app con los servicios backend, incluyendo la base de datos y las APIs, lo que permite que los usuarios reciban notificaciones inmediatas en caso de pérdida de conexión. Esta funcionalidad asegura que los usuarios puedan actuar rápidamente si experimentan problemas de conectividad.

Posteriormente, se realizó un análisis exhaustivo del rendimiento de la carga de datos en la app. Se identificaron áreas críticas, como consultas a la base de datos innecesariamente complejas, que afectaban la velocidad y eficiencia del proceso. Para solucionar esto, se optimizaron las consultas, simplificando las que eran demasiado complejas y aplicando técnicas de caching para reducir la cantidad de solicitudes al servidor. Esto resultó en una mejora significativa en la velocidad de carga de los datos, haciendo la app más eficiente.

Tras implementar estas mejoras, se realizaron pruebas exhaustivas para validar los cambios. Se verificó que el sistema de login fuera más seguro y eficiente, que las alertas funcionaran correctamente y que el rendimiento de la carga de datos hubiera mejorado. Los resultados fueron positivos: la app ofreció una experiencia más fluida y segura para los usuarios, con tiempos de respuesta optimizados y una mayor estabilidad en la autenticación.

Finalmente, se documentaron todos los resultados obtenidos durante las pruebas, incluyendo los cambios realizados, los resultados de las pruebas y recomendaciones para futuras actualizaciones. Esta documentación facilitará el mantenimiento de la app y permitirá realizar mejoras adicionales en el futuro.

Imágenes relacionadas:

```

1  const express = require("express");
2  const jwt = require("jsonwebtoken");
3  const bcrypt = require("bcrypt");
4  const cors = require("cors");
5
6  const app = express();
7  app.use(express.json());
8  app.use(cors());
9
10 const SECRET_KEY = "tu_clave_secreta"; // Usa una variable de entorno en producción
11
12 // Simulación de usuario almacenado en una BD (contraseña cifrada)
13 const usuarioDB = {
14   usuario: "admin",
15   clave: bcrypt.hashSync("123456", 10), // La clave almacenada está cifrada
16 };
17
18 // Endpoint de login
19 app.post("/login", async (req, res) => {
20   const { usuario, clave } = req.body;
21
22   // Validar si el usuario existe
23   if (usuario !== usuarioDB.usuario) {
24     return res.status(401).json({ mensaje: "Usuario o contraseña incorrectos" });
25   }
26
27   // Comparar la contraseña ingresada con la almacenada
28   const validPassword = await bcrypt.compare(clave, usuarioDB.clave);
29   if (!validPassword) {
30     return res.status(401).json({ mensaje: "Usuario o contraseña incorrectos" });
31   }

```

Figura 7: Código ejemplo



Figura 8: Fotografía de trabajo realizado sobre el desarrollo de alertas.

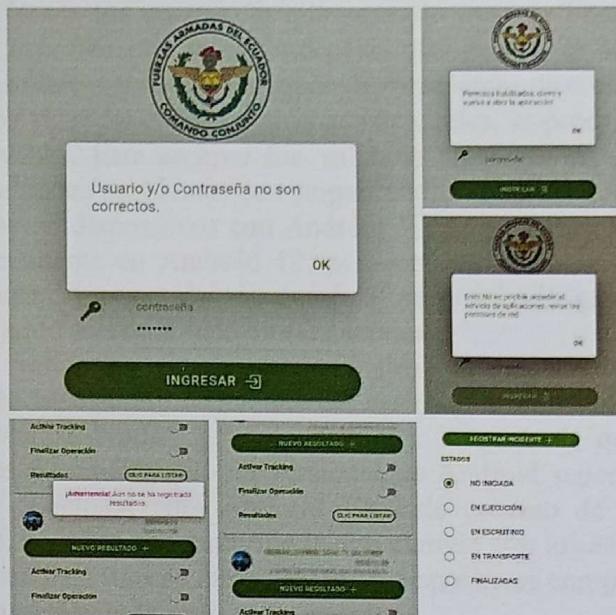


Figura 9. Collage de pruebas realizadas tras la implementación de mejoras.

Fase 5: Pruebas de funcionalidad y ajustes en la app móvil

Durante la fase de pruebas de funcionalidad y ajustes en la app móvil, el enfoque principal fue evaluar el rendimiento de la aplicación, corregir errores detectados tras la actualización de Android Studio, optimizar el código y garantizar la compatibilidad de la app con diferentes dispositivos y versiones de Android.

El proceso comenzó con una revisión detallada de los problemas comunes reportados tras la actualización de Android Studio. Se investigó si la nueva versión del entorno había generado conflictos con librerías o configuraciones anteriores. Esta investigación permitió identificar incompatibilidades y fallos, lo que resultó en la creación de un plan de acción para realizar los ajustes necesarios en las áreas críticas de la app, asegurando su correcto funcionamiento en el nuevo entorno de desarrollo.

A continuación, se realizó un análisis exhaustivo del código fuente, identificando redundancias y áreas de mejora. Se optimizó el código eliminando secciones innecesarias y refactorizando funciones para mejorar el rendimiento y la legibilidad. Además, se configuró el entorno de desarrollo en Android Studio, asegurando que todas las dependencias estuvieran correctamente instaladas y que la app fuera compatible con las versiones del SDK requeridas. Esta fase de optimización del código y configuración del entorno sentó las bases para futuras implementaciones dentro del sistema.

Después, se llevaron a cabo ajustes en la configuración de la app en Android Studio. Durante este proceso, se validaron las configuraciones iniciales del proyecto, corrigiendo conflictos derivados de configuraciones previas y realizando cambios en los parámetros clave para mejorar la compatibilidad y el rendimiento. Este paso también implicó la validación incremental de cada ajuste realizado, verificando que la app no presentara errores nuevos y mantuviera la estabilidad.

En la parte siguiente, se realizaron múltiples intentos de generar el archivo APK. Sin embargo, debido a problemas con dependencias obsoletas y configuraciones incorrectas, la mayoría de los intentos de compilación fueron fallidos. No obstante, se

logró corregir varios de los errores, lo que permitió avanzar hacia la obtención del APK funcional. La corrección de dependencias obsoletas y la actualización de las librerías necesarias fueron clave para resolver los problemas de compilación.

Tras la verificación final del código y ajustes adicionales, se procedió a la generación del APK para pruebas. Este archivo fue validado en dispositivos Android, lo que permitió detectar algunos problemas de compatibilidad entre versiones de Android. En particular, se probaron dispositivos con Android 12, 13 y 14, encontrando que la app funcionaba correctamente en Android 12, pero presentaba fallos de rendimiento en versiones superiores. A pesar de los desafíos, se consiguió generar el APK para Android 12 y se validó su estabilidad en esta versión.

Finalmente, se realizaron pruebas adicionales en dispositivos con diversas configuraciones de hardware y versiones de Android, con el objetivo de validar la funcionalidad y compatibilidad general de la app. A pesar de que la app presentó algunos problemas de rendimiento en versiones de Android superiores, se logró una versión estable y funcional para Android 12. Este proceso de pruebas y ajustes, aunque desafiante, permitió asegurar que la app cumpla con los estándares de calidad esperados y que esté lista para su lanzamiento en dispositivos compatibles.

Fase 6: Documentación de la aplicación móvil.

En esta fase se dedicó a la revisión general y documentación final del proyecto de la aplicación móvil. Se identificaron los aspectos clave del desarrollo para incluir en la documentación, como las decisiones tomadas, tecnologías utilizadas, y los desafíos superados. La documentación se estructuró en secciones principales: planificación del proyecto, desarrollo técnico, implementaciones clave, pruebas y validaciones, y conclusiones con recomendaciones. Se integraron diagramas, capturas de pantalla y esquemas para facilitar la comprensión. Al final, se completó y organizó un informe detallado y preciso que refleja el trabajo realizado durante las prácticas.

Fase 7: Optimización de la infraestructura de redes.

Durante la fase de optimización de la infraestructura de redes, se realizó un análisis exhaustivo de la red existente en un departamento específico, identificando áreas de mejora para optimizar la conectividad y el rendimiento. Se elaboró un inventario de los dispositivos de red, se evaluó el estado de los cables UTP y se realizaron pruebas de conectividad para identificar problemas. Además, se detectaron necesidades de mejora, como la actualización de configuraciones IP, reemplazo de cables dañados y ajustes en la configuración de dispositivos de red.

El día siguiente, se implementaron mejoras en la infraestructura de red, como la asignación de nuevas direcciones IP, la mejora de la conectividad entre equipos mediante la configuración adecuada de parámetros de red, y el reemplazo de cables UTP deteriorados. Finalmente, se realizaron pruebas de validación para asegurar que las mejoras lograran los objetivos de optimización, resultando en una red más eficiente y estable, con una mejor calidad de señal y mayor estabilidad en la transmisión de datos.

Fase 8: Configuración del router.

En la Fase 8 de configuración avanzada del router, se realizaron ajustes clave para mejorar la seguridad y el rendimiento de la red. Primero, se actualizó el firmware del router a la última versión, lo que mejoró su estabilidad y seguridad. Luego, se configuraron políticas avanzadas de firewall para filtrar el tráfico y bloquear accesos

no autorizados, fortaleciendo la protección contra posibles ataques. Además, se segmentó la red, creando una VLAN separada para dispositivos invitados, asegurando que los dispositivos externos no pudieran acceder a recursos internos sensibles. Finalmente, se realizaron pruebas de validación para confirmar que las configuraciones no afectaran negativamente la conectividad, asegurando que la red estuviera optimizada para un uso seguro y eficiente.

Fase 9: Presentación de la documentación del aplicativo móvil.

En la fase final del proyecto, se presentó la documentación del aplicativo móvil, detallando las mejoras implementadas, herramientas utilizadas y resultados obtenidos. Se realizó una revisión exhaustiva del informe para garantizar su precisión, claridad y cumplimiento de las normas APA, asegurando que las imágenes y diagramas estuvieran correctamente etiquetados. Posteriormente, se entregó el informe a los responsables del proyecto, destacando las actualizaciones realizadas, mejoras en funcionalidad y rendimiento, pruebas de compatibilidad y optimizaciones en el código. Durante la sesión de preguntas y respuestas, se resolvieron dudas y se realizaron ajustes menores en la documentación con base en las sugerencias recibidas. Finalmente, se mejoró el informe final, dejando una guía clara y completa para futuros desarrollos o mantenimiento del aplicativo móvil.

4. CONCLUSIONES

A lo largo de estas prácticas preprofesionales, pude aplicar mis conocimientos y habilidades en el ámbito de las Tecnologías de la Información, logrando cumplir con los objetivos establecidos. El proyecto me permitió trabajar en la optimización de infraestructuras de red, el desarrollo y mejora de un aplicativo móvil, y la documentación técnica del proceso. Estas actividades no solo me ayudaron a consolidar mi formación académica, sino que también fortalecieron mi capacidad para enfrentar retos del mundo real.

De los aprendizajes más significativos que destaco:

1. Administración de infraestructuras tecnológicas: Realicé configuraciones de red que mejoraron la conectividad y el rendimiento de los sistemas.
2. Desarrollo de aplicaciones móviles: Aprendí a actualizar y optimizar un aplicativo móvil, desde la revisión del código hasta la generación del APK y su validación.
3. Gestión de calidad: Implementé pruebas funcionales y de compatibilidad que garantizaron el correcto funcionamiento de la aplicación y su calidad final.

Siendo así, estas prácticas fueron una experiencia enriquecedora que me permitió poner en práctica mis conocimientos, aprender nuevas técnicas y adaptarme a las necesidades específicas de un entorno profesional, preparándome mejor para futuros desafíos.

5. RECOMENDACIONES

Para la institución:

- Continuar promoviendo proyectos que permitan a los estudiantes participar en procesos reales de desarrollo y optimización tecnológica, brindando acceso a herramientas actualizadas y documentación técnica detallada.

Para la universidad:

- Mantener el enfoque práctico de la carrera, incorporando más ejercicios reales que permitan a los estudiantes afrontar problemáticas similares a las del mercado laboral.

- Actualizar regularmente los contenidos relacionados con redes y desarrollo de aplicaciones para estar alineados con las últimas tendencias tecnológicas.

Para futuros practicantes:

- Familiarizarse previamente con las herramientas y tecnologías que serán utilizadas durante las prácticas, ya que esto facilitará el proceso de adaptación.
- Documentar cada actividad realizada, ya que esto no solo mejora la comprensión del proyecto, sino que también facilita futuras referencias y evaluaciones.



LA DIRECCIÓN GENERAL DE OPERACIONES MULTIDOMINIO DEL CC.FF.AA.

Confiere el presente

Certificado

Al Sr. Herrera Araujo Alan Vladimir

Por haber culminado con éxito sus prácticas pre-profesionales no remuneradas, en el Departamento C3I2 del Comando Conjunto de las Fuerzas Armadas, con una duración de 258 horas, del 24 de septiembre de 2024 al 29 de noviembre de 2024.

Quito D.M., 29 de noviembre de 2024

Víctor Garzón Sierra
CAPITÁN DE CORBETA - IG
JEFE TIC'S C3I2

Santiago Morales Sánchez
TENIENTE CORONEL EMT AVC
JEFE DEPARTAMENTO C3I2

Amitar Alvear Landeta
GENERAL DE BRIGADA
DIRECTOR GENERAL DE OPERACIONES DEL CC.FF.AA.