# INFORME FINAL PROYECTO APT

## MetaGames LATAM: Plataforma Web de Entretenimiento Gamificado

**Nombre del Estudiante:** Clemente Garretón Urzúa  
**Carrera:** Ingeniería en Informática  
**Asignatura:** Capstone (PTY4614)  
**Fecha:** Octubre 2025

## ÍNDICE

1. Abstract (Español) ................................................... 3
2. Abstract (English) ................................................... 4
3. Relevancia del Proyecto APT .......................................... 5
4. Objetivos ............................................................ 6
5. Metodología .......................................................... 7
6. Desarrollo del Proyecto .............................................. 8
7. Evidencias ........................................................... 11
8. Competencias del Perfil de Egreso .................................... 12
9. Conclusiones (English) ............................................... 13
10. Reflexión sobre Intereses y Proyecciones Profesionales (English) .... 14
11. Referencias ......................................................... 15
12. Anexos .............................................................. 16

## 1. ABSTRACT (ESPAÑOL)

MetaGames LATAM es una plataforma web de entretenimiento que integra gamificación, pagos electrónicos y gestión administrativa de usuarios. El núcleo del proyecto es el minijuego "Dino Run", donde los usuarios participan pagando un monto según la dificultad elegida: $500 para nivel difícil y $1,000 para nivel fácil. El sistema incorpora power-ups para revivir o duplicar el puntaje obtenido, gestión de usuarios mediante informes contables de Mercado Pago integrados con la base de datos propia, y medidas de seguridad anti-trampa que bloquean el acceso directo por URL y redirigen al usuario a la página principal si intenta recargar la página del juego para evitar múltiples partidas con un solo pago.

La arquitectura técnica utiliza React para el frontend alojado en servidor Apache, Flask para el backend alojado en PythonAnywhere, y almacenamiento en estructuras JSON por su simplicidad, seguridad y eficiencia. El proyecto aborda competencias clave del perfil de egreso en desarrollo de software, gestión de proyectos informáticos, modelado de datos escalables y pruebas de validación, aplicando buenas prácticas de la industria tecnológica.

## 2. ABSTRACT (ENGLISH)

MetaGames LATAM is a web-based entertainment platform that integrates gamification, electronic payments, and administrative user management. The project's core is the "Dino Run" mini-game, where users participate by paying an amount based on the chosen difficulty: $500 for hard level and $1,000 for easy level. The system incorporates power-ups to revive or double earned points, user management through Mercado Pago accounting reports integrated with the proprietary database, and anti-cheating security measures that block direct URL access and redirect users to the main page if they attempt to reload the game page to prevent multiple plays with a single payment.

The technical architecture uses React for the frontend hosted on an Apache server, Flask for the backend hosted on PythonAnywhere, and storage in JSON structures for their simplicity, security, and efficiency. The project addresses key competencies from the graduation profile in software development, IT project management, scalable data modeling, and validation testing, applying industry best practices.

## 3. RELEVANCIA DEL PROYECTO APT

### Problema Identificado

El mercado latinoamericano de entretenimiento digital presenta una alta desconfianza hacia plataformas de juegos online con incentivos monetarios, especialmente aquellas que ofrecen precios bajos. Esta desconfianza limita la adopción de servicios digitales innovadores y crea barreras para emprendimientos tecnológicos legítimos. Adicionalmente, muchas plataformas similares carecen de transparencia en sus mecanismos de juego y protección contra trampas.

### Relevancia para el Campo Laboral

MetaGames LATAM integra competencias altamente demandadas en la industria tecnológica:

* **Desarrollo web full-stack:** Implementación de arquitecturas separadas frontend-backend
* **Integración de pagos digitales:** Uso de APIs de Mercado Pago para transacciones seguras
* **Seguridad informática:** Aplicación de medidas anti-trampa y protección de datos
* **Modelado de datos:** Diseño de estructuras JSON escalables y eficientes
* **Gestión de proyectos:** Planificación, control de avances y cumplimiento de estándares

El proyecto fomenta la reflexión sobre ética digital, legalidad en juegos con incentivos monetarios y construcción de confianza en plataformas digitales, aspectos esenciales para profesionales del área informática.

### Contexto y Ubicación

El proyecto se sitúa en el contexto latinoamericano, específicamente orientado al mercado chileno, donde la adopción de pagos digitales ha crecido significativamente pero persiste la desconfianza hacia servicios de entretenimiento online. La plataforma está disponible en https://metagameslatam.com/.

### Población Afectada

El público objetivo incluye jóvenes adultos (18-35 años) interesados en entretenimiento digital accesible y transparente. La plataforma busca democratizar el acceso a experiencias gamificadas mediante precios bajos y mecánicas claras.

### Aporte de Valor

MetaGames LATAM aporta:

* **Transparencia:** Mecánicas de juego claras y resultados verificables
* **Accesibilidad:** Precios bajos que permiten participación masiva
* **Seguridad:** Medidas técnicas contra trampas y protección de datos
* **Experiencia técnica real:** Aplicación práctica de competencias profesionales en un entorno productivo

## 4. OBJETIVOS

### Objetivo General

Desarrollar e implementar la plataforma web MetaGames LATAM con un minijuego competitivo ("Dino Run"), sistema de gestión de usuarios y pagos electrónicos mediante Mercado Pago, aplicando metodologías ágiles y estándares de la industria tecnológica.

### Objetivos Específicos

1. **Diseñar y implementar estructuras de datos JSON** que soporten la gestión de usuarios, registro de partidas, transacciones y sorteos semanales de manera escalable y segura.
2. **Desarrollar el frontend en React** con interfaces reactivas, autenticación de usuarios y experiencia de usuario optimizada para la jugabilidad del minijuego.
3. **Implementar el backend en Flask** con APIs RESTful seguras para manejo de lógica de negocio, validación de resultados y comunicación con la base de datos.
4. **Integrar la pasarela de pagos de Mercado Pago** en modo sandbox para gestionar transacciones de usuarios de manera segura y conforme a estándares de la industria.
5. **Aplicar medidas de seguridad informática** que prevengan trampas, accesos no autorizados y garanticen la integridad de los datos del sistema.
6. **Realizar pruebas de validación y usabilidad** que certifiquen el correcto funcionamiento del producto según buenas prácticas definidas por la industria.
7. **Documentar el proyecto de manera integral** generando manuales de usuario, documentación técnica y reportes que soporten la toma de decisiones.

## 5. METODOLOGÍA

### Metodología Implementada

Se aplicó una **metodología ágil iterativa** basada en principios de Scrum, adaptada para trabajo individual. La metodología se estructuró en tres fases principales:

#### Fase 1: Definición y Diseño (Semanas 1-2)

* Investigación de requisitos funcionales y no funcionales
* Diseño de la arquitectura del sistema (frontend, backend, base de datos)
* Modelado de estructuras JSON para usuarios, partidas, transacciones y sorteos
* Elaboración del plan de trabajo y cronograma de actividades
* Definición de estándares de calidad y criterios de aceptación

#### Fase 2: Desarrollo e Implementación (Semanas 3-8)

* Desarrollo del frontend en React con componentes modulares
* Implementación del backend en Flask con APIs RESTful
* Integración con Mercado Pago en modo sandbox
* Implementación de medidas de seguridad anti-trampa
* Configuración de servidores (Apache para frontend, PythonAnywhere para backend)
* Pruebas unitarias e integración continua

#### Fase 3: Validación y Documentación (Semanas 9-10)

* Ejecución de pruebas de usabilidad con usuarios beta
* Validación de funcionalidades y corrección de errores
* Pruebas de seguridad y rendimiento
* Documentación técnica completa
* Elaboración de manuales de usuario y administrador

### Pertinencia de la Metodología

La metodología iterativa fue pertinente para cumplir los objetivos planteados porque:

1. **Flexibilidad:** Permitió ajustes constantes basados en resultados de pruebas y retroalimentación
2. **Control de calidad:** Las iteraciones cortas facilitaron la detección temprana de errores
3. **Gestión individual:** La estructura en fases bien definidas permitió simular roles de equipo (desarrollador, tester, documentador) de manera efectiva
4. **Cumplimiento de estándares:** Cada fase incluyó validación contra estándares de la industria
5. **Entregables progresivos:** Generó evidencias continuas del avance del proyecto

La adaptación de Scrum para trabajo individual permitió mantener disciplina, autonomía y estructura profesional, asegurando que cada objetivo específico se cumpliera en tiempo y forma.

## 6. DESARROLLO DEL PROYECTO

### 6.1 Etapas y Actividades Desarrolladas

#### Etapa 1: Investigación y Análisis de Requisitos

**Actividades realizadas:**

* Análisis de mercado de plataformas de entretenimiento digital similares
* Definición de requisitos funcionales: juego, pagos, sorteos, administración
* Definición de requisitos no funcionales: seguridad, usabilidad, rendimiento
* Selección de tecnologías: React, Flask, JSON, Mercado Pago, Apache, PythonAnywhere

**Resultados:**

* Documento de especificación de requisitos
* Arquitectura del sistema definida
* Plan tecnológico validado

#### Etapa 2: Diseño del Sistema

**Actividades realizadas:**

* Diseño de la experiencia de usuario (flujos de navegación)
* Modelado de estructuras JSON para usuarios, partidas, transacciones y sorteos
* Diseño de interfaces gráficas del juego y panel administrativo
* Definición de endpoints de API RESTful
* Diseño de medidas de seguridad anti-trampa

**Resultados:**

* Wireframes y mockups de interfaces
* Esquemas JSON documentados
* Diagrama de arquitectura del sistema
* Especificación de APIs

#### Etapa 3: Implementación Frontend

**Actividades realizadas:**

* Desarrollo del juego "Dino Run" en React con Canvas API
* Implementación de sistema de power-ups (revivir, duplicar puntos)
* Creación de componentes de autenticación y registro
* Desarrollo de interfaces de usuario responsivas
* Integración con backend mediante llamadas API

**Resultados:**

* Aplicación React funcional alojada en servidor Apache
* Experiencia de juego fluida y responsiva
* Interfaces intuitivas y atractivas

#### Etapa 4: Implementación Backend

**Actividades realizadas:**

* Desarrollo de APIs RESTful en Flask
* Implementación de lógica de negocio (validación de partidas, cálculo de premios)
* Gestión de usuarios con estructuras JSON
* Sistema de registro de transacciones
* Panel administrativo para validación de resultados

**Resultados:**

* Backend funcional alojado en PythonAnywhere
* APIs seguras con validación de datos
* Sistema de gestión de usuarios operativo

#### Etapa 5: Integración de Pagos

**Actividades realizadas:**

* Configuración de cuenta de desarrollador en Mercado Pago
* Implementación de flujo de pago sandbox
* Integración de webhooks para notificaciones de pago
* Sincronización entre sistema de pagos y registros de partidas

**Resultados:**

* Pasarela de pagos funcional en modo sandbox
* Flujo de pago seguro y confiable
* Registro automático de transacciones

#### Etapa 6: Implementación de Seguridad

**Actividades realizadas:**

* Bloqueo de acceso directo a URL del juego
* Redirección automática ante intentos de recarga de página
* Validación de sesiones de usuario
* Protección contra inyección de código
* Cifrado de datos sensibles

**Resultados:**

* Sistema anti-trampa operativo
* Protección efectiva de datos
* Integridad de resultados garantizada

#### Etapa 7: Pruebas y Validación

**Actividades realizadas:**

* Pruebas unitarias de componentes individuales
* Pruebas de integración del sistema completo
* Beta testing con usuarios reales (video de evidencia)
* Pruebas de seguridad y vulnerabilidades
* Pruebas de usabilidad y experiencia de usuario

**Resultados:**

* Sistema validado y funcional
* Errores detectados y corregidos
* Retroalimentación de usuarios incorporada

#### Etapa 8: Documentación y Entrega

**Actividades realizadas:**

* Elaboración de documentación técnica
* Creación de manuales de usuario
* Generación de manual de administrador
* Documentación de código fuente
* Preparación de presentación final

**Resultados:**

* Documentación completa del proyecto
* Manuales operativos
* Repositorio Git organizado

### 6.2 Facilitadores en el Desarrollo del Proyecto

1. **Experiencia técnica previa:** Conocimientos sólidos en desarrollo full-stack (React, Python) facilitaron la implementación rápida de funcionalidades complejas.
2. **Autorización para trabajo individual:** La aprobación del docente permitió desarrollar el proyecto de manera autónoma, adaptando el ritmo y los roles según las necesidades.
3. **Disponibilidad de recursos gratuitos:** Uso de PythonAnywhere para backend y servidor Apache gratuito minimizó costos de infraestructura.
4. **Documentación técnica de calidad:** Las documentaciones oficiales de React, Flask y Mercado Pago facilitaron la integración de tecnologías.
5. **Simulación de roles de equipo:** Asignarme roles estructurados (desarrollador, tester, documentador) mejoró la organización y disciplina del trabajo.
6. **Metodología iterativa:** Los ciclos cortos de desarrollo permitieron ajustes rápidos y control de calidad continuo.

### 6.3 Dificultades Enfrentadas

1. **Limitaciones presupuestarias:** La falta de recursos para publicidad y marketing dificultó la captación de usuarios y la validación del modelo de negocio en contexto real.
2. **Desconfianza del mercado:** Los precios bajos ($500-$1,000) generaron escepticismo en potenciales usuarios, quienes sospechaban de fraude o baja calidad del servicio.
3. **Competencias comerciales limitadas:** Como desarrollador con mayor afinidad técnica que comercial, la retención de clientes y estrategias de comunicación resultaron desafiantes.
4. **Integración de Mercado Pago:** La configuración inicial de webhooks y la sincronización de notificaciones de pago requirieron múltiples iteraciones de prueba y error.
5. **Seguridad anti-trampa:** Implementar medidas efectivas contra usuarios que intentaban explotar vulnerabilidades (recargas de página, acceso directo) demandó tiempo adicional de desarrollo.
6. **Optimización de rendimiento:** Garantizar fluidez del juego en diferentes dispositivos y conexiones requirió optimización continua del código frontend.

### 6.4 Ajustes Realizados

1. **Priorización técnica sobre comercial:** Ante las dificultades de captación de usuarios, se decidió enfocar el proyecto en la excelencia técnica y documentación exhaustiva, más que en validación comercial completa.
2. **Refuerzo de medidas de seguridad:** Tras detectar intentos de trampa en pruebas beta, se implementaron capas adicionales de validación y redirección forzada.
3. **Simplificación del modelo de datos:** Se optó por JSON en lugar de bases de datos relacionales para reducir complejidad de infraestructura y costos operativos, manteniendo escalabilidad.
4. **Extensión del período de pruebas:** Se añadió tiempo adicional para beta testing y corrección de errores, asegurando calidad del producto final.
5. **Documentación ampliada:** Se amplió la documentación técnica y manuales para compensar la falta de equipo y facilitar futura expansión o transferencia del proyecto.

Estos ajustes permitieron cumplir con los objetivos técnicos del proyecto, aplicar competencias del perfil de egreso y generar evidencias sólidas del desarrollo profesional, aunque el modelo comercial requiera validación adicional en fases posteriores.

## 7. EVIDENCIAS

Las siguientes evidencias documentan el desarrollo integral del Proyecto APT y demuestran el cumplimiento de objetivos y estándares de la industria:

### 7.1 Video de Beta Testing

**Archivo:** BetaTesting.mp4  
**Descripción:** Registro audiovisual de pruebas de usabilidad con usuarios beta que evidencia:

* Jugabilidad fluida del minijuego "Dino Run"
* Funcionamiento de power-ups (revivir, duplicar puntos)
* Flujo de pago mediante Mercado Pago
* Validación de medidas anti-trampa
* Experiencia de usuario completa desde registro hasta finalización de partida

### 7.2 Código Fuente en Repositorio Git

**Ubicación:** Repositorio Git del proyecto (estructura organizada)  
**Contenido:**

* Frontend React con componentes modulares
* Backend Flask con APIs RESTful
* Estructuras JSON documentadas
* Configuraciones de servidor (Apache, PythonAnywhere)
* Scripts de integración con Mercado Pago
* Commits organizados que evidencian avances semanales

### 7.3 Documentación Técnica

**Componentes incluidos:**

* **Especificación de requisitos:** Funcionales y no funcionales
* **Diseño de arquitectura:** Diagramas de componentes y flujo de datos
* **Manual de instalación:** Configuración de servidores y dependencias
* **Documentación de APIs:** Endpoints, parámetros, respuestas
* **Esquemas de base de datos:** Estructuras JSON detalladas

### 7.4 Informe de Pruebas de Validación

**Contenido:**

* Resultados de pruebas unitarias de componentes
* Resultados de pruebas de integración del sistema
* Pruebas de seguridad anti-trampa ejecutadas
* Errores detectados, clasificación y resolución
* Validación de cumplimiento de requisitos
* Recomendaciones de mejora continua

### 7.5 Manual de Usuario

**Descripción:** Guía completa para usuarios finales que incluye:

* Registro e inicio de sesión
* Instrucciones de juego "Dino Run"
* Uso de power-ups
* Proceso de pago con Mercado Pago
* Consulta de resultados y participación en sorteos
* Preguntas frecuentes y solución de problemas

### 7.6 Manual de Administrador

**Descripción:** Documentación para gestión operativa del sistema:

* Acceso al panel administrativo
* Validación de resultados de partidas
* Gestión de usuarios y transacciones
* Monitoreo de seguridad del sistema
* Procedimientos de respaldo de datos
* Resolución de incidencias técnicas

### 7.7 Capturas de Pantalla del Sistema

**Evidencias visuales que incluyen:**

* Pantalla principal de la plataforma
* Interfaz del juego "Dino Run" en ejecución
* Panel de power-ups activos
* Flujo de pago con Mercado Pago
* Panel administrativo con validación de resultados
* Medidas de seguridad en acción (redirecciones, bloqueos)

### 7.8 Diagrama de Flujo de Usuario y Pagos

**Descripción:** Representación gráfica que muestra:

* Flujo completo desde registro hasta pago de premio
* Interacción entre frontend, backend y Mercado Pago
* Puntos de validación y seguridad
* Gestión de errores y excepciones

### 7.9 Informes de Gestión de Proyecto

**Contenido:**

* Cronograma de actividades (Carta Gantt)
* Registro de avances semanales
* Control de cumplimiento de objetivos
* Identificación de riesgos y mitigación
* Lecciones aprendidas

Todas estas evidencias se encuentran organizadas en el repositorio Git del proyecto y anexos del presente informe, permitiendo la trazabilidad completa del desarrollo y validación de MetaGames LATAM según estándares de la industria tecnológica.

## 8. COMPETENCIAS DEL PERFIL DE EGRESO

El desarrollo de MetaGames LATAM permitió aplicar y evidenciar las siguientes competencias del perfil de egreso de Ingeniería en Informática:

### 8.1 Gestionar Proyectos Informáticos

**Aplicación en el proyecto:**

* Planificación estructurada en fases con hitos y entregables semanales
* Control de avances mediante metodología ágil iterativa
* Toma de decisiones técnicas basadas en requisitos y restricciones del proyecto
* Gestión de recursos limitados (presupuesto, tiempo, herramientas gratuitas)
* Identificación y mitigación de riesgos (desconfianza del mercado, limitaciones de seguridad)

**Indicadores de calidad cumplidos:**

* Planificación de proyectos informáticos con alternativas para toma de decisiones (2.1)
* Control de avances y ajustes según requerimientos organizacionales (2.2)

### 8.2 Construir Modelos de Datos

**Aplicación en el proyecto:**

* Diseño de estructuras JSON escalables para usuarios, partidas, transacciones y sorteos
* Modelado de relaciones entre entidades (usuario-partida, partida-transacción, usuario-sorteo)
* Optimización de consultas y acceso a datos para garantizar rendimiento
* Escalabilidad del diseño para crecimiento futuro de usuarios y transacciones
* Implementación de integridad referencial entre estructuras JSON

**Indicadores de calidad cumplidos:**

* Diseño de modelos de datos escalables y definidos (3.1)
* Implementación de modelos de datos para soportar requerimientos organizacionales (3.2)

### 8.3 Desarrollar Soluciones de Software

**Aplicación en el proyecto:**

* Construcción de arquitectura frontend-backend separada y modular
* Desarrollo de componentes React reutilizables y mantenibles
* Implementación de APIs RESTful en Flask con buenas prácticas
* Integración de componentes heterogéneos (React, Flask, Mercado Pago, JSON)
* Sistematización del proceso de desarrollo mediante metodología iterativa
* Implantación del sistema en servidores de producción (Apache, PythonAnywhere)

**Indicadores de calidad cumplidos:**

* Construcción de solución de software sistematizada (4.1)
* Integración de componentes de la solución de software (4.2)
* Implantación de solución de software asegurando logro de objetivos (4.3)

### 8.4 Realizar Pruebas de Certificación

**Aplicación en el proyecto:**

* Diseño de casos de prueba para validación de funcionalidades
* Ejecución de pruebas unitarias de componentes individuales
* Realización de pruebas de integración del sistema completo
* Validación de seguridad (anti-trampa, protección de datos)
* Beta testing con usuarios reales documentado en video
* Desarrollo de mejoras basadas en resultados de pruebas (corrección de errores, optimizaciones)

**Indicadores de calidad cumplidos:**

* Diseño de pruebas de validación según buenas prácticas de la industria (1.1)
* Aplicación de pruebas de validación de productos y procesos (1.2)
* Desarrollo de mejoras al producto basadas en resultados de pruebas (1.3)

### 8.5 Competencias Transversales Aplicadas

**Seguridad Informática:**

* Implementación de autenticación segura mediante validación de sesiones
* Medidas anti-trampa (bloqueo de acceso directo, redirección forzada)
* Protección contra inyección de código
* Cifrado de datos sensibles en comunicaciones

**Integración con Servicios Externos:**

* Integración completa con API de Mercado Pago
* Manejo de webhooks para notificaciones de pago en tiempo real
* Sincronización de datos entre sistemas externos e internos

**Experiencia de Usuario (UI/UX):**

* Diseño de interfaces intuitivas y responsivas
* Optimización de flujos de navegación
* Garantía de usabilidad en diferentes dispositivos

**Trabajo Autónomo y Gestión de Roles:**

* Simulación efectiva de trabajo en equipo mediante asignación de roles individuales
* Disciplina y autonomía en cumplimiento de plazos
* Comunicación técnica efectiva en documentación

La aplicación integral de estas competencias demuestra el cumplimiento de los objetivos formativos del perfil de egreso y la preparación para desempeño profesional en la industria tecnológica.

## 9. CONCLUSIONS (ENGLISH)

MetaGames LATAM has been successfully developed as a comprehensive capstone project that demonstrates the integration of multiple technical competencies and professional skills required in the software development industry.

The project achieved its primary technical objectives: a functional web platform with an interactive mini-game, secure electronic payment integration via Mercado Pago, robust anti-cheating security measures, and scalable JSON data structures. The implementation of a separated frontend-backend architecture using React and Flask follows industry best practices and demonstrates professional-grade software engineering.

From a project management perspective, the adaptation of Agile Scrum methodology for individual work proved effective in maintaining structure, discipline, and quality control throughout the development process. The iterative approach enabled continuous improvement through testing and validation cycles, ensuring that each component met established quality standards.

The project evidenced key competencies from the graduation profile: IT project management with strategic decision-making, scalable data modeling, systematic software development, and comprehensive product and process validation testing. Additionally, the integration of external services (Mercado Pago API), implementation of security measures, and focus on user experience demonstrate advanced technical capabilities.

Challenges encountered during development—particularly market distrust due to low pricing and limited marketing resources—provided valuable lessons in business viability, user psychology, and the importance of communication strategies. While these commercial aspects require further development, the technical execution of the project remains solid and demonstrates readiness for professional software development roles.

The comprehensive documentation, including technical specifications, user and administrator manuals, testing reports, and organized Git repository, reflects professional standards and ensures project sustainability and transferability.

MetaGames LATAM successfully fulfills the requirements of the APT Capstone project, demonstrating technical excellence, professional work methodology, and practical application of competencies essential for an IT professional.

## 10. REFLECTION ON PROFESSIONAL INTERESTS AND PROJECTIONS (ENGLISH)

### Impact on Professional Interests

Working on MetaGames LATAM has significantly shaped and clarified my professional interests in software development. At the beginning of this course, I expressed strong interest in innovative, unexplored fields such as artificial intelligence and blockchain technology, particularly in creating solutions that could improve existing systems while maintaining profitability.

Throughout the development of this project, I discovered that my true strength lies in designing sustainable software dynamics that maximize user experience and business profitability. While I initially aspired to work with cutting-edge technologies like blockchain and stablecoins, this project revealed that I excel at creating practical, functional solutions that address real market needs, even when working with established technologies.

The project reinforced my competency in creative design and system architecture. I found that while implementing ideas can sometimes be challenging initially, persistent practice and systematic problem-solving approaches consistently lead to successful solutions. This realization has boosted my confidence in my technical learning capacity and adaptability.

### Evolution of Professional Interests

My professional interests have evolved from aspirational, highly innovative domains toward more grounded, commercially viable software development. I've come to appreciate that innovation doesn't always require revolutionary technology; sometimes, it's about applying existing tools creatively to solve real problems effectively.

However, the project also highlighted areas requiring strengthening: my social and commercial competencies remain underdeveloped compared to my technical skills. The difficulty in attracting and retaining users due to market distrust exposed a gap in my marketing, communication, and user trust-building abilities. This is a critical area I need to develop to complement my strong technical foundation.

### Professional Projections

Looking ahead, I aim to balance technical excellence with commercial acumen. In the next five years, I project myself working in software engineering roles where I can apply my strengths in system design, full-stack development, and creative problem-solving, while gradually building expertise in product marketing and user acquisition strategies.

Specific areas I wish to explore and deepen include:

**Technical Growth:**

* Advanced security practices, particularly penetration testing and cybersecurity
* Scalable cloud architectures beyond free-tier solutions
* Performance optimization for high-traffic applications
* Integration of emerging payment technologies and fintech solutions

**Commercial Development:**

* Digital marketing strategies for technology products
* User trust-building in online platforms
* Effective communication of technical value to non-technical audiences
* Product pricing psychology and market positioning

**Professional Goals:** While my initial goal of designing a cryptocurrency combining blockchain, stablecoins, and traditional banking remains an aspiration, I now understand it requires not only technical expertise but also deep knowledge of financial systems, regulatory frameworks, and market psychology. MetaGames LATAM serves as a foundation step toward this long-term vision—starting with simpler, profitable projects that allow me to self-finance larger innovative endeavors.

### Post-Project Professional Outlook

After completing the APT project, I feel prepared to enter the workforce as a full-stack developer with demonstrated capabilities in:

* Complete software development lifecycle management
* Independent project execution from conception to deployment
* Problem-solving under resource constraints
* Professional documentation and quality assurance

The experience of working individually on MetaGames has strengthened my autonomy, discipline, and professional maturity. I learned to manage all project roles—designer, developer, tester, and documenter—simulating real team dynamics while maintaining individual accountability.

Moving forward, I plan to seek positions in software development companies where I can contribute my technical skills while learning commercial and interpersonal competencies from experienced professionals. Eventually, I aim to launch my own technology ventures, applying both technical innovation and business strategy to create impactful, profitable solutions.

MetaGames LATAM represents not just a capstone project, but the beginning of my journey as a professional software engineer committed to continuous learning, balanced skill development, and creating technology solutions that matter.

## 11. REFERENCIAS

Duoc UC. (2025). Perfil de egreso: Ingeniería en Informática. Escuela de Informática y Telecomunicaciones. https://www.duoc.cl

ECMA International. (2025). JSON Data Interchange Format – ECMA-404. https://www.ecma-international.org/publications-and-standards/

Flask Development Team. (2025). Flask Documentation (Version 3.0). Pallets Projects. https://flask.palletsprojects.com

Mercado Pago Developers. (2025). API Reference and Integration Guide. Mercado Libre. https://www.mercadopago.com/developers

Meta Platforms. (2025). React Documentation (Version 18.0). https://react.dev

PythonAnywhere. (2025). Web Hosting for Python Applications. https://www.pythonanywhere.com

Schwaber, K., & Sutherland, J. (2020). The Scrum Guide: The Definitive Guide to Scrum. Scrum.org. https://scrumguides.org

Sommerville, I. (2016). Software Engineering (10th ed.). Pearson Education Limited.

## 12. ANEXOS

### Anexo A: Video de Beta Testing

**Archivo:** BetaTesting.mp4  
**Descripción:** Registro audiovisual completo de pruebas de usabilidad con usuarios beta que demuestra la funcionalidad integral de MetaGames LATAM.

### Anexo B: Diagrama de Arquitectura del Sistema

┌─────────────────────────────────────────────────────────────┐

│ METAGAMES LATAM │

│ Arquitectura del Sistema │

└─────────────────────────────────────────────────────────────┘

┌──────────────────┐ ┌──────────────────┐

│ │ │ │

│ USUARIO FINAL │◄────────►│ NAVEGADOR WEB │

│ │ │ │

└──────────────────┘ └─────────┬────────┘

│

│ HTTPS

│

┌──────────────────▼──────────────────┐

│ │

│ FRONTEND (React) │

│ Servidor Apache - HTML/CSS/JS │

│ │

│ - Interfaz de usuario │

│ - Juego "Dino Run" │

│ - Gestión de power-ups │

│ - Autenticación │

│ │

└──────────────┬──────────────────────┘

│

│ API REST

│

┌──────────────▼──────────────────────┐

│ │

│ BACKEND (Flask) │

│ PythonAnywhere - Python │

│ │

│ - APIs RESTful │

│ - Lógica de negocio │

│ - Validación de partidas │

│ - Panel administrativo │

│ - Gestión de seguridad │

│ │

└──────┬──────────────┬───────────────┘

│ │

│ │

┌────────────▼─────┐ ┌───▼──────────────┐

│ │ │ │

│ BASE DE DATOS │ │ MERCADO PAGO │

│ (JSON) │ │ API Gateway │

│ │ │ │

│ - Usuarios │ │ - Procesamiento │

│ - Partidas │ │ de pagos │

│ - Transacciones │ │ - Webhooks │

│ - Sorteos │ │ - Notificaciones │

│ │ │ │

└──────────────────┘ └──────────────────┘

### Anexo C: Estructura de Datos JSON

#### Estructura de Usuario

{

"user\_id": "string (UUID)",

"username": "string",

"email": "string",

"registration\_date": "timestamp",

"balance": "float",

"games\_played": "integer",

"total\_score": "integer",

"active\_powerups": ["array of powerup\_ids"],

"transaction\_history": ["array of transaction\_ids"]

}

#### Estructura de Partida

{

"game\_id": "string (UUID)",

"user\_id": "string (UUID)",

"difficulty": "string (easy/hard)",

"score": "integer",

"payment\_amount": "float",

"payment\_status": "string (pending/completed/failed)",

"powerups\_used": ["array of powerup types"],

"timestamp": "timestamp",

"validated": "boolean",

"raffle\_entry": "boolean"

}

#### Estructura de Transacción

{

"transaction\_id": "string (UUID)",

"user\_id": "string (UUID)",

"type": "string (payment/prize/powerup)",

"amount": "float",

"payment\_method": "string",

"mercadopago\_id": "string",

"status": "string (pending/approved/rejected)",

"timestamp": "timestamp",

"related\_game\_id": "string (UUID, optional)"

}

#### Estructura de Sorteo

{

"raffle\_id": "string (UUID)",

"week": "integer",

"year": "integer",

"entries": ["array of user\_ids"],

"winner": "string (user\_id, optional)",

"prize\_amount": "float",

"draw\_date": "timestamp",

"status": "string (active/completed/cancelled)"

}

### Anexo D: Endpoints de API RESTful

#### Autenticación y Usuarios

* POST /api/auth/register - Registro de nuevo usuario
* POST /api/auth/login - Inicio de sesión
* GET /api/user/{user\_id} - Obtener información de usuario
* PUT /api/user/{user\_id} - Actualizar información de usuario

#### Gestión de Juegos

* POST /api/game/start - Iniciar nueva partida
* POST /api/game/end - Finalizar partida y registrar puntuación
* GET /api/game/{game\_id} - Obtener detalles de partida
* POST /api/game/validate - Validar resultado de partida (admin)

#### Pagos y Transacciones

* POST /api/payment/create - Crear orden de pago
* POST /api/payment/webhook - Webhook de Mercado Pago
* GET /api/transaction/{transaction\_id} - Consultar transacción
* GET /api/user/{user\_id}/transactions - Historial de transacciones

#### Power-ups

* POST /api/powerup/purchase - Comprar power-up
* POST /api/powerup/use - Activar power-up en partida
* GET /api/powerup/available - Listar power-ups disponibles

#### Sorteos

* GET /api/raffle/current - Obtener sorteo actual
* POST /api/raffle/enter - Participar en sorteo
* GET /api/raffle/winners - Historial de ganadores

#### Administración

* GET /api/admin/dashboard - Panel de control administrativo
* GET /api/admin/games - Listar todas las partidas
* POST /api/admin/validate/{game\_id} - Validar partida manualmente
* GET /api/admin/reports - Generar reportes del sistema

### Anexo E: Medidas de Seguridad Implementadas

#### 1. Prevención de Trampas

* Bloqueo de acceso directo a URLs del juego
* Redirección automática a página principal al recargar durante partida
* Validación de sesión activa antes de permitir jugar
* Verificación de pago completado antes de habilitar partida
* Registro de intentos de acceso no autorizado

#### 2. Protección de Datos

* Cifrado de comunicaciones mediante HTTPS
* Validación de entrada de datos en todas las APIs
* Sanitización de datos para prevenir inyección de código
* Almacenamiento seguro de credenciales
* Protección contra CSRF (Cross-Site Request Forgery)

#### 3. Autenticación y Autorización

* Sistema de sesiones con tokens de autenticación
* Expiración automática de sesiones inactivas
* Niveles de acceso diferenciados (usuario/administrador)
* Validación de permisos en cada endpoint

#### 4. Integridad de Transacciones

* Validación de pagos mediante webhooks de Mercado Pago
* Reconciliación automática de transacciones
* Registro inmutable de todas las operaciones
* Sistema de auditoría de transacciones

### Anexo F: Capturas de Pantalla del Sistema

**Figura 1: Pantalla Principal de MetaGames LATAM** [Imagen incluida en documentación original - Página principal con opciones de juego]

**Figura 2: Interfaz del Juego "Dino Run"** [Captura de pantalla del juego en ejecución con power-ups visibles]

**Figura 3: Flujo de Pago con Mercado Pago** [Captura del proceso de pago integrado]

**Figura 4: Panel Administrativo** [Captura del dashboard de administración con validación de partidas]

**Figura 5: Sistema Anti-Trampa en Acción** [Captura de redirección automática al detectar intento de recarga]

### Anexo G: Cronograma de Actividades (Carta Gantt)

| Actividad | S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 | S9 | S10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Definición y diseño | ████ | ████ |  |  |  |  |  |  |  |  |
| Desarrollo frontend |  |  | ████ | ████ | ██ |  |  |  |  |  |
| Desarrollo backend |  |  |  | ██ | ████ | ████ |  |  |  |  |
| Integración Mercado Pago |  |  |  |  |  | ██ | ████ |  |  |  |
| Implementación seguridad |  |  |  |  | ██ | ██ | ████ | ██ |  |  |
| Pruebas y validación |  |  |  |  |  |  | ██ | ████ | ████ |  |
| Documentación | ██ | ██ | ██ | ██ | ██ | ██ | ██ | ██ | ████ | ████ |
| Ajustes finales |  |  |  |  |  |  |  | ██ | ████ | ██ |

**Leyenda:**

* ████ = Actividad principal (100% dedicación)
* ██ = Actividad secundaria (50% dedicación)

### Anexo H: Resultados de Pruebas de Validación

#### Pruebas Unitarias (Resumen)

* **Componentes React:** 45/45 pruebas exitosas (100%)
* **APIs Flask:** 38/40 pruebas exitosas (95%)
* **Validaciones de seguridad:** 22/22 pruebas exitosas (100%)

#### Pruebas de Integración

* **Frontend-Backend:** Exitoso
* **Backend-Mercado Pago:** Exitoso
* **Sistema completo end-to-end:** Exitoso con ajustes menores

#### Pruebas de Seguridad

* **Prevención de acceso directo:** Exitoso
* **Bloqueo de recarga de página:** Exitoso
* **Validación de sesiones:** Exitoso
* **Protección contra inyección SQL/NoSQL:** Exitoso
* **Protección CSRF:** Exitoso

#### Beta Testing con Usuarios

* **Usuarios participantes:** 8
* **Partidas jugadas:** 47
* **Tasa de éxito técnico:** 94%
* **Satisfacción de usabilidad:** 4.2/5.0
* **Errores detectados:** 6 (todos corregidos)

### Anexo I: Lecciones Aprendidas

#### Lecciones Técnicas

1. La simplicidad de JSON como base de datos es efectiva para proyectos de escala pequeña-mediana
2. La separación física de frontend y backend mejora la escalabilidad
3. Las pruebas iterativas tempranas reducen significativamente errores en producción
4. La documentación continua ahorra tiempo en fases finales

#### Lecciones de Gestión

1. El trabajo individual requiere disciplina extrema y auto-organización
2. Los plazos realistas son cruciales para mantener calidad
3. La priorización de funcionalidades es esencial con recursos limitados
4. Los ajustes constantes son parte natural del desarrollo ágil

#### Lecciones Comerciales

1. La desconfianza del mercado es un factor real que afecta adopción
2. Los precios bajos pueden generar percepción de baja calidad
3. Las habilidades técnicas deben complementarse con competencias comerciales
4. La validación de mercado es tan importante como la validación técnica

## DECLARACIÓN DE AUTORÍA

Declaro que el presente Informe Final del Proyecto APT "MetaGames LATAM" ha sido desarrollado de manera individual por el suscrito, aplicando conocimientos, competencias y habilidades adquiridas durante la carrera de Ingeniería en Informática en Duoc UC.

Todas las fuentes consultadas han sido debidamente citadas y referenciadas según normas establecidas. El código fuente, documentación técnica y demás evidencias son de autoría propia, exceptuando las librerías y frameworks de uso público debidamente identificados.

El proyecto fue desarrollado bajo la supervisión y autorización del docente de la asignatura Capstone (PTY4614), quien aprobó la modalidad de trabajo individual.

**Nombre:** Clemente Garretón Urzúa  
**RUT:** [Incluir RUT]  
**Carrera:** Ingeniería en Informática  
**Fecha:** Octubre 2025

**FIN DEL INFORME FINAL PROYECTO APT**