



FACULTAD DE INGENIERIAS Y CIENCIAS AGROPECUARIAS

IMPLEMENTACIÓN DE UN PROTOTIPO DE SISTEMA INTEGRADO
MULTIPLATAFORMA PARA LA CONSULTA DE CARTELERAS,
INFORMACIÓN Y COMPRA DE BOLETOS DE LAS PRINCIPALES CADENAS
DE CINES DEL ECUADOR.

Trabajo de titulación presentado en conformidad con los requisitos establecidos
para optar por el título de Ingeniero Electrónico en Redes de Información.

Profesor guía

Ing. Walter Efraín Navarro Rubio

Autores

Sebastián Mansfield Rojas

Javier Alejandro Granda García

Año

2014

DECLARACIÓN DEL PROFESOR GUÍA

“Declaro haber dirigido este trabajo a través de reuniones periódicas con el (los) estudiante(s), orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación. “

Ing. Walter Efraín Navarro Rubio
Ing. En Sistemas de Computación e Informática
C.I. 171185843-9

DECLARACION DE AUTORÍA DEL ESTUDIANTE

Declaro (amos) que este trabajo es original, de mi (nuestra) autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.

Sebastián Mansfield Rojas

030212488-8

Javier Alejandro Granda García

1104092836

AGRADECIMIENTOS

Agradecemos a Palapa y a sus integrantes por su tiempo y colaboración en ayuda a cumplir este trabajo de titulación.

Agradecemos al Ing. Walter Efraín Navarro por guiarnos en el proceso de la elaboración de este proyecto.

RESUMEN

El presente trabajo de titulación se centra en el diseño, desarrollo e implementación de un prototipo de sistema integrado multiplataforma para la consulta de carteleras, información y compra de boletos de las principales cadenas de cines del Ecuador. Realizado con el marco de trabajo de Microsoft, MSF, y la metodología de desarrollo ágil, SCRUM. Se implementó una arquitectura cliente-servidor (remoto), tomando en cuenta criterios de diseño como la alta disponibilidad, escalabilidad, facilidad de administración y experiencia de uso. Adicionalmente, se optimizó el tiempo de desarrollo e implementación al incorporar técnicas de programación híbridas de aplicaciones móviles. El proyecto se desarrolló durante un período de aproximadamente 4 meses y medio, en el cual, se lograron los objetivos propuestos y la implementación deseada en las plataformas web, web-móvil, iOS y Android.

ABSTRACT

This graduation project focuses on the design, development and implementation of a multi-platform integrated prototype system for consulting movie times, information and ticket purchasing major theater chains of Ecuador. Made with the MSF Microsoft framework, and the agile development methodology, SCRUM. A client-server (remote) architecture was implemented, taking into account design criteria such as high availability, scalability, manageability, and user experience. Additionally, the time of development and implementation was optimized due to the incorporation of hybrid programming techniques for mobile applications. The project was developed over a period of about 4 and a half months, in which the objectives and desired implementation in web, mobile-web, iOS and Android platforms were achieved.

ÍNDICE

Introducción	1
Antecedentes	1
Marco Referencial	2
Alcance	3
Objetivos	5
Objetivo General	5
Objetivos Específicos	6
1. Marco Teórico	6
1.1. Metodología	6
1.1.1. Microsoft Solutions Framework (MSF)	7
1.1.2. Scrum.....	9
1.2. Conceptos	13
1.2.1. HTML5	13
1.2.2. JavaScript	14
1.2.3. CSS.....	14
1.2.4. SASS.....	15
1.3. Marcos de Trabajo y Herramientas	15
1.3.1. Web Clásica	15
1.3.1.1. Python – Django.....	15
1.3.2. Móvil.....	18
1.3.2.1. JAVA.....	18
1.3.2.2. Objective C.....	19
1.3.2.3. Sencha Touch	19
1.3.2.4. Compass	23
1.3.2.5. Phonegap.....	23
1.3.3. Servidor y Bases de Datos.....	26
1.3.3.1. Amazon Linux AMI	26
1.3.3.2. Amazon Elastic Compute Cloud (Amazon EC2)	26

1.3.3.3. Amazon Relational Database Service (Amazon RDS).....	27
1.3.3.4. Amazon Simple Storage Service (Amazon S3).....	27
1.3.3.5. Servidor HTTP Apache	28
1.3.3.6. WSGI (Web Server Gateway Interface)	28
1.3.3.7. MySQL	28
1.3.3.8. Alternative PHP Cache (APC).....	29
2. Desarrollo del sistema	29
2.1. Visión	29
2.1.1. Beneficios y limitaciones.	30
2.1.2. Matriz de riesgos	33
2.1.3. Cronograma General.....	34
2.1.4. Historias de Usuario	35
2.2. Planeación.....	43
2.2.1 Equipo de trabajo	43
2.2.1. Lista de Producto	44
2.3. Desarrollo	47
2.3.1. Criterios de Diseño.....	47
2.3.2. Infraestructura física.....	48
2.3.3. Diagrama de arquitectura física y lógica	50
.....	50
2.3.4. Modelo EER	51
2.3.5. Modelo físico de base de datos.....	52
2.3.6. Tabla inventario arquitectura MVC (Modelo-Vista-Controlador).....	53
2.3.6.1. Aplicación.....	53
2.3.6.2. Web.....	54
2.3.7. Funcionalidad	55
2.3.7.1. Módulo Películas	55
2.3.7.2. Módulo Cines	56
2.3.7.3. Módulo Perfil	57
2.3.7.4. Horarios.....	57
2.3.7.5. Compra de boletos	58

2.3.8. Sprint Cero	59
2.3.8.1. Objetivo	59
2.3.8.2. Alcance	59
2.3.8.3. Lista de Sprint	60
2.3.8.4. Cierre y retrospectiva	61
2.3.9. Primer Sprint	62
2.3.9.1. Objetivo	62
2.3.9.2. Alcance	62
2.3.9.3. Lista de Sprint	63
2.3.9.4. Demos.....	65
2.3.9.5. Cierre y Retrospectiva.....	66
2.3.10. Segundo Sprint.....	67
2.3.10.1. Objetivo	67
2.3.10.2. Alcance	67
2.3.10.3. Lista de Sprint	68
2.3.10.4. Demos.....	70
2.3.10.5. Cierre y Retrospectiva.....	72
2.3.11. Tercer Sprint	72
2.3.11.1. Objetivo	72
2.3.11.2. Alcance	72
2.3.11.3. Lista de Sprint	73
2.3.11.4. Demos.....	74
2.3.11.5. Cierre y Retrospectiva.....	76
2.3.12. Cuarto Sprint	76
2.3.12.1. Objetivo	76
2.3.12.4. Demos.....	78
2.3.12.5. Cierre y Retrospectiva.....	80
2.3.13. Quinto Sprint	80
2.3.13.1. Objetivo	80
2.3.13.2. Alcance	80
2.3.13.4. Demos.....	82
2.3.13.5. Cierre y Retrospectiva.....	84

2.3.14. Sexto Sprint.....	84
2.3.14.1. Objetivo	84
2.3.14.2. Alcance	84
2.3.14.4. Demos.....	86
2.3.14.5. Cierre y Retrospectiva.....	88
2.3.15. Séptimo Sprint.....	88
2.3.15.1. Objetivo	88
2.3.15.2. Alcance	88
2.3.15.3. Lista de Sprint	89
2.3.15.4. Demos.....	90
2.3.15.5. Cierre y Retrospectiva.....	93
2.3.16. Octavo Sprint.....	93
2.3.16.1. Objetivo	93
2.3.16.2. Alcance	93
2.3.16.3. Lista de Sprint	94
2.3.16.4. Demos.....	95
2.3.16.5. Cierre y Retrospectiva.....	98

3. Implementación y Pruebas..... 99

3.1. Implementación..... 99

3.1.1. Back-End.....	99
3.1.1.1. Instalación de Actualizaciones de Linux, Ajuste de zonas horarias	99
3.1.1.2. Instalación de PHP y extensiones del mismo.....	99
3.1.1.3. Instalación de Apache Httpd	100
3.1.1.4. Instalación de Mysql.....	100
3.1.1.5. Instalación de Modulo PHP APC.....	100
3.1.1.6. Instalación de Modulo Mod-WSGI.....	101
3.1.1.7. Configuraciones PHP	101
3.1.1.7. Iniciar servicios Apache en el servidor	102
3.1.1.8. Configuración de base de datos de producción	102

3.2. Pruebas 102

3.2.1. Sprint de Pruebas.....	103
3.2.1.1. Objetivo	103
3.2.1.2. Alcance	103
3.2.1.3. Lista de Sprint	104
3.2.1.5 Cierre y retrospectiva	106
3.2.2. Pruebas no funcionales.....	107
3.2.2.1. Prueba de carga.....	107
3.2.2.2. Prueba de carga por URLs.....	108
3.2.2.3. Prueba de tiempo de carga por tipos de contenido.....	108
3.2.2.4. Métricas de servidor EC2 y base de datos RDS	109
4. Conclusiones y Recomendaciones	110
4.1. Conclusiones	110
4.2. Recomendaciones.....	112
5. REFERENCIAS:.....	114

Introducción

Antecedentes

La industria digital en la última década ha experimentado un crecimiento exponencial sostenido, debido principalmente a la participación del internet como una herramienta de trabajo y medio de comunicación que por su naturaleza tiene prestaciones que se ajustan con el ritmo de desarrollo globalizado del mundo en la actualidad.

A medida del avance en la inclusión de las TIC (Tecnologías de la Información y Comunicación) en el Ecuador, cada vez más personas utilizan el internet como fuente de información en sus actividades diarias. Especialmente el segmento joven de la población.

Los últimos cinco años en nuestro país han sido los más significativos en cuanto a las estadísticas de uso y tráfico del internet desde teléfonos inteligentes y dispositivos móviles. Según la información obtenida por el Instituto Nacional de Estadísticas y Censos (INEC), en la encuesta “Tecnologías de la Comunicación e Información”, en el 2012 el 12.2% de las personas que tenían un celular poseían un teléfono inteligente, esto frente al 8.4% registrado en 2011. La tendencia de los últimos cuatro años ha sido que la población que se encuentra entre los 16 y 24 años presenta el mayor uso del internet con el 64.9%, seguido del 46.2% que son la población entre 25 y 34 años.

Las cadenas más importantes de cines del Ecuador están enfocando sus recursos y estrategias de crecimiento hacia el área digital web y móvil, considerando que el crecimiento de estos medios genera una importante ventana para llegar al usuario final. La cadena Multicines tiene en producción

una aplicación para visualizar sus carteleras y la información de las películas. Las demás cadenas de cines mantienen su página web como su mejor herramienta para acercarse a los medios digitales.

El éxito de este tipo de sistemas y sus aplicaciones móviles han sido comprobados y verificados mediante casos de éxito reales como Fandango en los Estados Unidos de Norteamérica que según Appannie, empresa dedicada a la generación de métricas y rankings de aplicaciones móviles reconocida mundialmente, sitúa a Fandango en la 10ma posición en aplicaciones de entretenimiento en la tienda iTunes de Estados Unidos.

A nivel mundial en el 2012, según un estudio realizado por la empresa Cisco, el tráfico móvil se incrementó un 70% y el uso de smartphones en el mundo un 81 %. Según Cisco para el final del 2013 se espera que la cantidad de dispositivos móviles conectados al internet excedan al número de habitantes en la tierra.

Marco Referencial

En el Ecuador no existe en la actualidad un sistema integrado de sitios web y aplicaciones móviles mediante los cuales revisar la información de los complejos de cines e interactuar con tu comunidad sobre las películas.

El usuario debe ingresar a la página web de cada complejo para poder ver los horarios, lo que genera un problema de movilidad y accesibilidad al momento de revisar la información de más de un establecimiento, por lo tanto, se crea la oportunidad de desarrollar una solución nueva e innovadora, que minimice los problemas de movilidad al estar disponible en los teléfonos inteligentes y, de accesibilidad al integrar la información de todas las salas de cine relevantes en un solo formato.

Consideramos que éste es el momento más oportuno para desarrollar aplicaciones móviles, que den una solución real y efectiva a las problemáticas

diarias en el Ecuador debido a la creciente tendencia global y local del uso de internet y dispositivos móviles.

La innovación tecnológica que se propone a través de este proyecto es diseñar un sistema que en todas sus plataformas se priorice la experiencia del usuario mediante las tendencias de uso actuales en los diferentes dispositivos en los que correrá la aplicación; utilizando una metodología de desarrollo de aplicaciones móviles basada en marcos de trabajo (Framework) que optimizará el tiempo de implementación en las diferentes plataformas y maximizará el alcance de la compatibilidad del sistema.

La arquitectura de datos del sistema estará diseñada e implementada en la nube teniendo en cuenta la escalabilidad, centralización y facilidad de administración, todo esto con el objetivo principal de asegurar el mayor porcentaje de disponibilidad del sistema en cualquier circunstancia.

Mediante un servidor RDS (Relational Database System) se implementará el motor de base de datos MySQL, los servidores web funcionarán en un servidor Linux EC2 (Elastic Compute Cloud) y el repositorio de imágenes estará alojado en un servidor S3 (Simple Storage Services). Todos estos servicios son prestados por la empresa Amazon en su gama de productos y servicios AWS (Amazon Web Services).

Todas las tecnologías detrás de la implementación multiplataforma del sistema serán debidamente investigadas para asegurar su calidad.

Alcance

En este proyecto se realizará el diseño, desarrollo e implementación de un prototipo del sistema integrado multiplataforma que pretende unificar, simplificar y agilizar las consultas de carteleras y compra de boletos de las principales cadenas de cines del Ecuador, y toda la información relacionada con las películas que éstas distribuyen.

El sistema será implementado en las siguientes plataformas:

- Aplicación Nativa iOS.
- Aplicación Nativa Android.
- Aplicación Web para exploradores que soporten Webkit.
- Sitio Web con administración desarrollado en el Web Framework Python –Django.

En el sistema se desarrollarán las siguientes funcionalidades:

- Visualización de películas (tráiler, sinopsis, duración, género, elenco, director, censura) y horarios que se encuentren en los siguientes períodos de tiempo:
 - Día de la consulta:
 - Semana de la consulta
- Visualización de películas (tráiler, sinopsis, duración, género, elenco, director, censura) y horarios que se encuentren en cartelera en el transcurso de la semana de la consulta.
- Visualización de películas (tráiler, sinopsis, duración, género, elenco, director, censura) y horarios en cartelera de cada uno de los complejos de cines.
- Filtro de horarios de funciones para el día de la consulta, el día siguiente o dos días después dentro de cada complejo o película.
- Visualización de funciones (película y complejo) dentro de las próximas dos horas.
- Utilización del GPS del dispositivo móvil para desplegar películas y cines dentro de la ciudad en la que te encuentres.

- Selección de la ciudad en caso de que el GPS no esté disponible guardando las preferencias de locación automáticamente.

Las siguientes funcionalidades estarán disponibles en un módulo de perfil después de que el usuario se haya registrado en la aplicación mediante su cuenta de Facebook:

- Compartir películas con amigos en las redes sociales más importantes.
- Calificar y compartir la calificación de una película en un rango de 1 a 5 estrellas.
- Visualizar la calificación de una película basada en calificaciones anteriores hechas por usuarios del sistema.
- Guardar y visualizar los cines favoritos para un rápido acceso a los mismos.
- Guardar, compartir y visualizar los estrenos esperados.

Mediante estas funcionalidades se dará un valor agregado al usuario que contará con un medio social para expresar y compartir sus gustos e interactuar con su comunidad, generando un mayor valor a través del tiempo.

Objetivos

Objetivo General

Desarrollar un prototipo de sistema integrado multiplataforma para la consulta de carteleras, información y compra de boletos de las principales cadenas de cines del Ecuador.

Objetivos Específicos

- Identificar los requerimientos necesarios de acuerdo con las necesidades del entorno que se presentan en la actualidad.
- Diseñar una interfaz de usuario basado en las tendencias de uso de dispositivos móviles e internet de la actualidad.
- Integrar los diferentes lenguajes de cliente (JavaScript, HTML, CSS) y plataformas de servidor (PHP, Django, Apache, MySQL) en la que estará desarrollado el sistema para optimizar recursos y tiempo.
- Adecuar la información de las carteleras de los complejos de cine para mostrarla unificada al usuario final
- Integrar herramientas sociales en el sistema mediante el API (Application Programming Interface) de Facebook
- Realizar pruebas al prototipo desarrollado para asegurar la calidad y eficacia de sus funcionalidades.

1. Marco Teórico

1.1. Metodología

La metodología de desarrollo ágil es un conjunto de principios, herramientas y prácticas que tienen como objetivo construir y desarrollar software de la manera más rápida y eficientemente posible asegurando la total funcionalidad

del mismo. Gracias a la estrecha colaboración de equipos multidisciplinarios y organizados por sí mismos, conjuntamente con los dueños del producto se logra desarrollar iterativa e incrementalmente partes de un producto final que satisfaga completamente las necesidades del cliente.

Una de las principales ventajas de un marco de trabajo ágil, es la flexibilidad para incorporar nuevos requerimientos y para adaptarse a los cambios, así como, la atención al detalle y la calidad que se logran a través de la retroalimentación constante entre todos los miembros del equipo. Hoy en día, cada vez más empresas y equipos dedicados a la industria del software han tomado los marcos de desarrollo ágil como el estándar de facto para sus proyectos, porque brinda una plataforma de comunicación más abierta y asegura resultados encapsulados en tiempos controlables y analizables.

1.1.1. Microsoft Solutions Framework (MSF)

Según Microsoft Developer Network (2014) “Microsoft Solutions Framework (MSF) es un enfoque personalizable para entregar correcta y más rápidamente soluciones tecnológicas, con menos personas y menos riesgo, pero con resultados de más calidad. MSF ayuda a los equipos a resolver directamente las causas más comunes de error en el proyecto de tecnología, lo cual mejora los índices de buenos resultados, de calidad de la solución y de impacto comerciales.

MSF se centra en:

- Alinear objetivos empresariales y tecnológicos
- Establecer objetivos, roles y responsabilidades claros para el proyecto
- Implementar un proceso iterativo, basado en hitos/puntos de control
- Administrar riesgos de forma proactiva
- Respuestas efectivas a los cambios”

Este marco de trabajo será utilizado como referencia a seguir para mantener una organización en el proceso de desarrollo ágil.

Este framework propone una estructura a seguir que permiten validar la arquitectura, funcionalidad y procesos de manera eficiente y mantener al software propuesto siempre en constante mejora y desarrollo.

Según Microsoft Developer Network (2014) los principios fundamentales de MSF son:

- Fomentar comunicaciones abiertas
- Trabajar hacia una visión compartida
- Autorización para miembros del equipo
- Establecer responsabilidades claras y compartidas
- Entregar valor incremental
- Mantenerse ágil, atento a los cambios
- Invertir en calidad
- Aprender de todas las experiencias
- Asociarse con clientes internos y externos

Este marco de trabajo consiste básicamente de dos modelos, el modelo de equipo y el modelo de gobernanza.

En el Modelo de Equipo se compone de los siguientes roles:

- Administración de productos
- Administración del programa
- Arquitectura
- Desarrollo
- Experiencia del usuario
- Prueba
- Versión/operaciones

El Modelo de Gobernanza se lo puede observar en la figura uno:

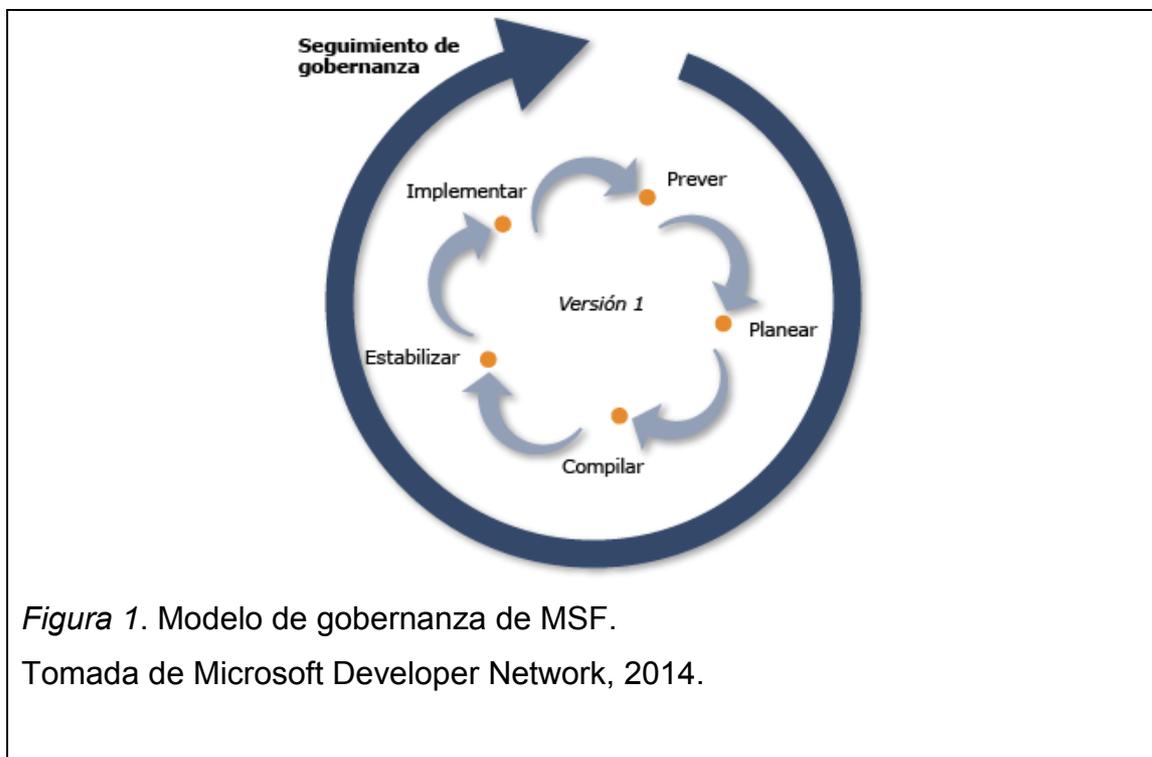


Figura 1. Modelo de gobernanza de MSF.

Tomada de Microsoft Developer Network, 2014.

1.1.2. Scrum

Según Scrum Alliance, s.f. b y Schwaber K. y Sutherland J., 2013 scrum es un marco de trabajo ágil, iterativo e incremental, mediante el cual se organiza y se estructura la colaboración entre equipos y personas de diferentes áreas que se encuentran trabajando para el desarrollo de un mismo software o producto.

Scrum se basa en la teoría de control de procesos empírica o empirismo para el desarrollo holístico de un producto. La toma de decisiones se realiza en base a lo previamente conocido.

Los pilares fundamentales del empirismo son: transparencia, inspección y adaptación.

El proceso de desarrollo con Scrum ocurre en pequeñas partes, donde cada pieza se basa en piezas creadas previamente. Esto permite que productos o proyectos de alta complejidad se puedan organizar y llevar a cabo de una manera más precisa y con la colaboración de todos los involucrados. Scrum proporciona un pequeño conjunto de reglas, que crean una estructura suficiente, para que los equipos puedan centrar su innovación en resolver problemas que de lo contrario podrían ser un desafío difícil de superar.

Objetivamente Scrum se compone de roles, eventos, instrumentos y reglas asociadas. Todos los componentes cumplen objetivos y son indispensables para asegurar la buena adopción y uso de Scrum. Como se puede ver en la figura 2, la iteración de trabajo por parte de cada miembro del equipo usando los instrumentos necesarios crea un proceso iterativo y ágil de desarrollo. En la figura dos se muestra el proceso iterativo de SCRUM.



Aunque este marco de trabajo ofrezca las herramientas y la estructura necesaria para desarrollar todo tipo de productos o software, sus principios son sumamente simples y, su base consiste principalmente de tres roles:

- **Dueño del Producto (Product Owner):**

Este individuo debe representar a todas las instituciones o personas que estén interesadas en el desarrollo y culminación del producto.

En otras palabras, el dueño del producto es la voz del cliente, el que se encarga y asegura que se cumplan los requerimientos y se maximice el valor del producto. Dentro de sus tareas también se encuentra la administración de la **Lista de Producto (Product Backlog)**, la cual consta de todos los requisitos y funcionalidades que podrían ser necesarios en el producto, y el trabajo del Equipo de Desarrollo, al cual se le podría asignar la gestión de la Lista de Producto.

- **Equipos de Desarrollo (Development Team):**

El Equipo de Desarrollo está conformado de profesionales de las diferentes ramas de la ciencia que son responsables de la construcción y la entrega de las partes incrementales y potencialmente operacionales del producto al final de cada **Sprint (Iteración incremental)**. Este equipo tiene la característica de ser auto-organizado y coordinador de su propio trabajo. En términos simples, el equipo convierte los elementos de la Lista de Producto en características funcionales tangibles.

- **Scrum Master:**

La tarea del Scrum Master es asegurar que el marco de trabajo es entendido y debidamente adoptado por todo el equipo Scrum, también, actúa como el facilitador transversal de los demás roles en caso de existir dificultades o conflictos, y, se asegura que todos los miembros del

equipo tengan las herramientas que necesitan para hacer su trabajo. Debe tener en cuenta las reglas, estructura y roles para garantizar que tanto los equipos como el producto trabajen y se desarrollen con la mínima cantidad de inconvenientes y retrasos. El Scrum Master es el canal de comunicación con los individuos externos al equipo Scrum.

Eventos de Scrum

Los eventos de Scrum son definidos con el objetivo de brindar canales de comunicación efectivos para analizar, aprender y mejorar en el proceso de desarrollo de un producto.

Sprint

El **Sprint** es el evento de eventos en Scrum, representa una unidad de desarrollo basada en un tiempo definido (1-4 semanas), en la cual se especifican funcionalidades del producto, que al final del tiempo previamente definido deben ser potencialmente puestas en producción por el equipo de desarrollo. Cada nuevo Sprint comienza inmediatamente después de la finalización del Sprint anterior.

La reunión para definir los requerimientos y funcionalidades para establecer el objetivo final del Sprint se llama **Reunión de Planificación de Sprint (Sprint Planning Meeting)**. La **Revisión del Sprint (Sprint Review)**, es la reunión mediante la cual se revisa si el objetivo se cumplió, y, si los requerimientos definidos al inicio se implementaron; finalmente la **Retrospectiva del Sprint (Sprint Retrospective)** es la reunión final de cada Sprint en la cual se evalúan los resultados y tomando como referencia el tiempo en el que se culminó el Sprint se analiza si se necesita hacer cambios de ser necesarios y cuales serían estos para mejorar los siguientes Sprints.

El **Daily Scrum** es una reunión que se realiza de pie durante 15 minutos al inicio de cada día de trabajo, con el objetivo de permitir un flujo constante de

comunicación entre los miembros del equipo. Durante esta reunión cada persona habla sobre el trabajo que ya realizó, en lo que se encuentra actualmente y en lo que planea trabajar a continuación, así como cualquier dificultad u obstáculo que tenga para ejecutar sus tareas.

Para el presente trabajo de titulación se escogió Scrum como marco de trabajo de desarrollo principalmente por sus características en cuanto a estructura, agilidad y adaptación al cambio. El hecho de estar desarrollando un prototipo de un producto totalmente nuevo y con herramientas de última generación nos plantea el desafío de tener que hacer cambios sobre la marcha y aun así poder avanzar de acuerdo a un tiempo estimado. Scrum encaja perfectamente en nuestros requerimientos ya que nos brinda la habilidad de adaptarnos a los cambios que suponen la investigación y la experimentación propias de desarrollo de software y poder organizar estos cambios de manera de que no retrasen nuestro trabajo. Evidentemente se hará una adaptación de Scrum en cuanto a roles ya que no se cuenta con un equipo completo.

1.2. Conceptos

1.2.1. HTML5

HTML5 es el más reciente estándar del lenguaje HTML, la versión anterior se publicó en 1999, por lo tanto, una actualización era necesaria, la implementación principal de sintaxis es que se desarrolló conjuntamente HTML4 con XHTML. Aunque su desarrollo se encuentre todavía en fase experimental, muchos desarrolladores web y navegadores han adoptado desde ya la nueva sintaxis y etiquetas, las cuales permiten crear aplicaciones web mucho más interactivas y dinámicas, sin necesidad de usar plug-ins adicionales como previamente se solía hacer. Otra de las ventajas importantes de este lenguaje es que es multi-plataforma, lo que permite que las aplicaciones corran tanto en un teléfono inteligente como en una tablet o computadora. HTML5

trabaja conjuntamente con CSS3, la tercera revisión del lenguaje de estilos CSS para la web. W3Schools, 2013.

1.2.2. JavaScript

JavaScript es un lenguaje de programación interpretado y dinámico, desarrollado inicialmente por Netscape y ahora propiedad de Sun Microsystems, es usado en millones de sitios web y aplicaciones de servidores en todo el mundo, y, brinda la capacidad de agregar interactividad a aplicaciones, principalmente en la web. Actualmente el estándar de JavaScript es ECMAScript, el cual es compatible con todos los navegadores, en especial la última versión 5.1 que desde 2012 fue adoptada por la mayoría de navegadores. JavaScript puede funcionar como un lenguaje procedimental pero mayoritariamente se lo utiliza como orientado a objetos. Sus capacidades dinámicas incluyen la construcción de objetos en tiempo de ejecución, listas de parámetros variables, etc. Mozilla Developer Network, s.f.

1.2.3. CSS

“Hojas de Estilo en Cascada (Cascading Style Sheets), es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos.

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier

cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento.” Word Wide Web Consortium, s.f.

1.2.4. SASS

Sass es una extensión de CSS que añade potencia y elegancia al lenguaje básico, permite el uso de variables, reglas anidadas, mixins (clase de la que hereda una subclase), y más, todo con una sintaxis totalmente compatible con CSS.

Sass ayuda a mantener grandes hojas de estilo bien organizadas, y a poner en producción rápidamente páginas de estilos pequeñas, sobre todo con la ayuda de la biblioteca de estilos Compass. Catlin, Weizenbaum, Eppstein, 2013.

1.3. Marcos de Trabajo y Herramientas

Se han seleccionado los siguientes marcos de trabajo para el desarrollo de las aplicaciones tanto web como móviles por sus prestaciones técnicas y de apertura de desarrollo, a continuación explicamos sus características.

1.3.1. Web Clásica

1.3.1.1. Python – Django

Según Python Organization, s.f. Python es un lenguaje de programación de código abierto dinámico y de alto nivel que es usado en una gran variedad aplicaciones debido a su gran robustez así como a su completitud en cuanto a funcionalidades en su librería estándar. Algunas de sus principales características incluyen:

- Tipos de datos dinámicos y de alto nivel

- Manejo de errores basado en excepciones
- Modularidad de paquetes
- Orientación a objetos intuitiva
- Sintaxis muy clara y entendible

Además de todas estas bondades, Python es muy flexible ya que se acopla de excelente manera con librerías de otros lenguajes como Java o .NET y es soportado por la mayoría de sistemas operativos en el mercado.

Según Holovaty y Kaplan-Moss, 2009. Cap.1 Django es un marco de trabajo (framework) gratuito y de código abierto para el desarrollo de aplicaciones Web basado en el lenguaje de programación de alto nivel Python.

Este proporciona abstracciones de alto nivel de los patrones comunes de desarrollo web, accesos directos para tareas de programación frecuentes y convenciones claras sobre la forma de resolver los problemas.

Django es parte de una nueva generación de frameworks Web en la industria, el enfoque MVC (Modelo-Vista-Controlador) separa el modelado del dominio, la presentación al usuario y las acciones que se basan en la interacción y entrada de datos por parte del usuario en estos tres elementos:

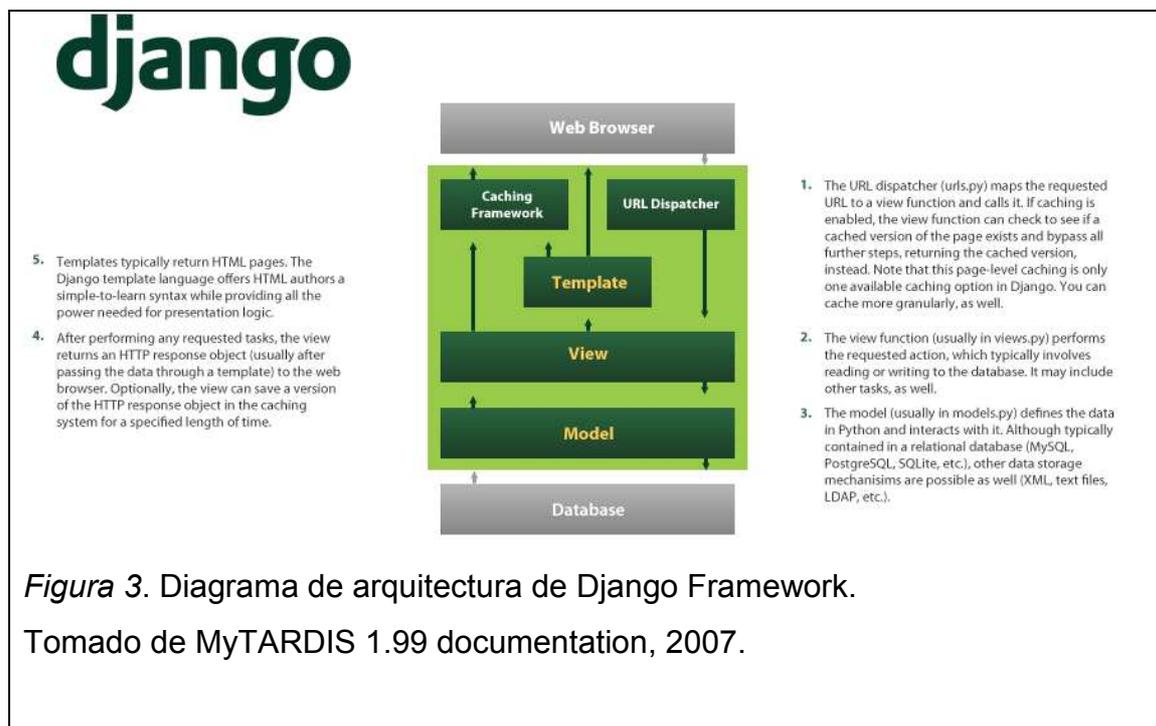
- **Modelo:** Gestiona el comportamiento de los datos y su estructura, actúa ante los pedidos de información que vienen normalmente del controlador.
- **Vista (Template en Django):** Esta maneja la presentación de la información al usuario (HTML, CSS y JavaScript).
- **Controlador (View en Django):** Actúa como intérprete de las acciones de entrada del usuario en la aplicación para que la vista actualice o cambie los datos del modelo. En esta capa se realiza el mapeo objeto-relacional que constituye el núcleo mismo de funcionamiento.

Adicionalmente Django implementa un despachador URL basado en expresiones regulares para manejar el direccionamiento de las diferentes plantillas.

Django hace énfasis en la reutilización y la conectividad de componentes, el rápido desarrollo y el principio de no repetirse (DRY, Don't Repeat Yourself).

Django también proporciona un procedimiento administrativo opcional para crear, leer, actualizar y eliminar. La interfaz de administración se genera de forma dinámica a través de los modelos de administración.

La figura 3 ilustra perfectamente la arquitectura de Django basada en el enfoque MVC.



Django presenta la oportunidad de poner en aplicación la última tecnología en frameworks de desarrollo web de alto nivel y así brindar al sistema un alto rendimiento en el manejo de los datos y la optimización del desarrollo evitando repetir la programación de componentes.

Adicional a todos los beneficios que por su tecnología y su infraestructura brinda Django, durante el proceso de desarrollo del sistema fue necesario proporcionar un sistema de administración y manejo del contenido que sea no solo amigable al usuario, sino lo suficientemente robusto para manejar la cantidad de datos que supone el sistema y sus funcionalidades.

Django nos pone a disposición un sistema de administración autogenerado en base a los modelos previamente establecidos, de esta forma optimizamos el tiempo de desarrollo y el código se enfoca esencialmente en la lógica del negocio y su presentación.

1.3.2. Móvil

Para hacer las aplicaciones móviles, en los sistemas operativos principales como iOS y Android, hemos seleccionado frameworks que sean ampliamente utilizados en el ambiente de desarrollo de aplicaciones móviles tanto por su flexibilidad y cantidad de herramientas como por su soporte en la comunidad.

1.3.2.1. JAVA

Java es un lenguaje de programación y plataforma de computación lanzado por Sun Microsystems en 1995. Actualmente existen miles de aplicaciones y sitios web que funcionan bajo este lenguaje. Java es rápido, seguro y fiable.

Desde ordenadores portátiles a centros de datos, desde consolas de juegos hasta supercomputadoras científicas, desde teléfonos móviles a Internet.

Para el propósito de este sistema JAVA es el lenguaje de programación principal con el que se desarrolla aplicaciones en Android. Oracle, 2013.

1.3.2.2. Objective C

Objective-C es el lenguaje de programación principal que se utiliza para programar software para OS X y iOS. Es una modificación o conjunto del lenguaje de programación C, proporcionando capacidades orientadas a objetos y un tiempo de ejecución dinámico.

Objective-C hereda la sintaxis, tipos primitivos, y el flujo de las instrucciones de control del lenguaje C y se agrega la sintaxis para la definición de clases y métodos. También añade soporte a nivel de idioma para gestión de gráfico de objetos y objetos literales mientras que proporciona escritura dinámica. Apple, 2013.

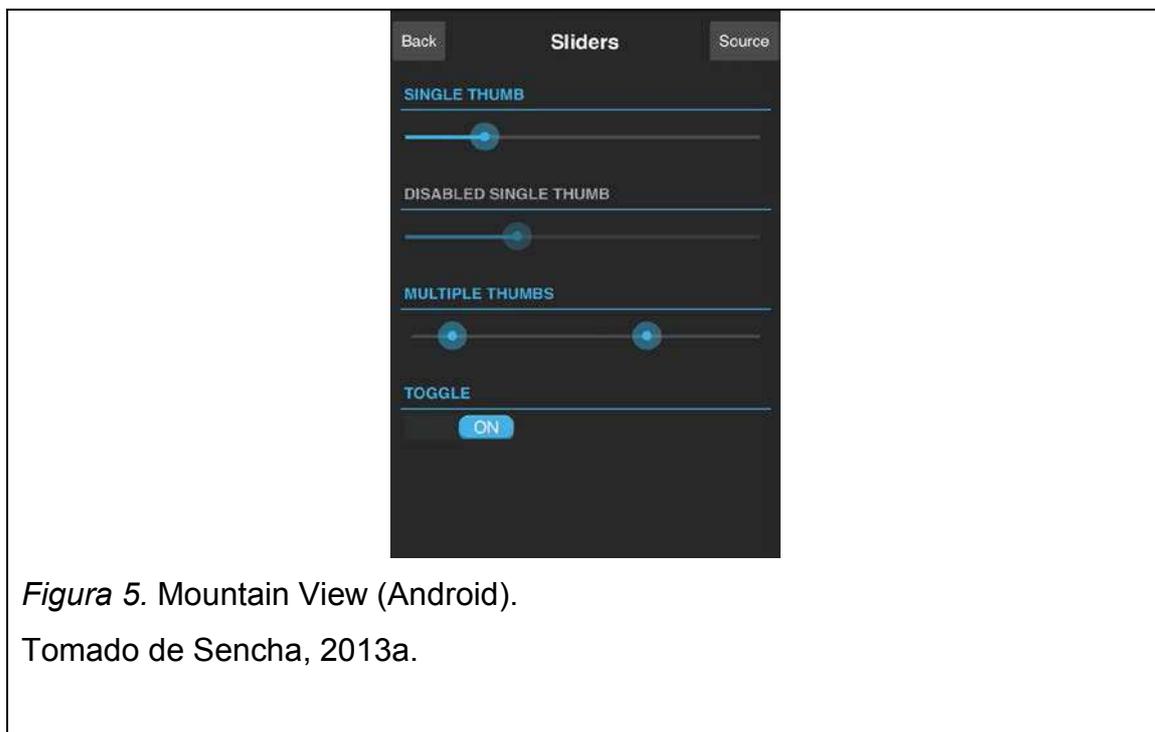
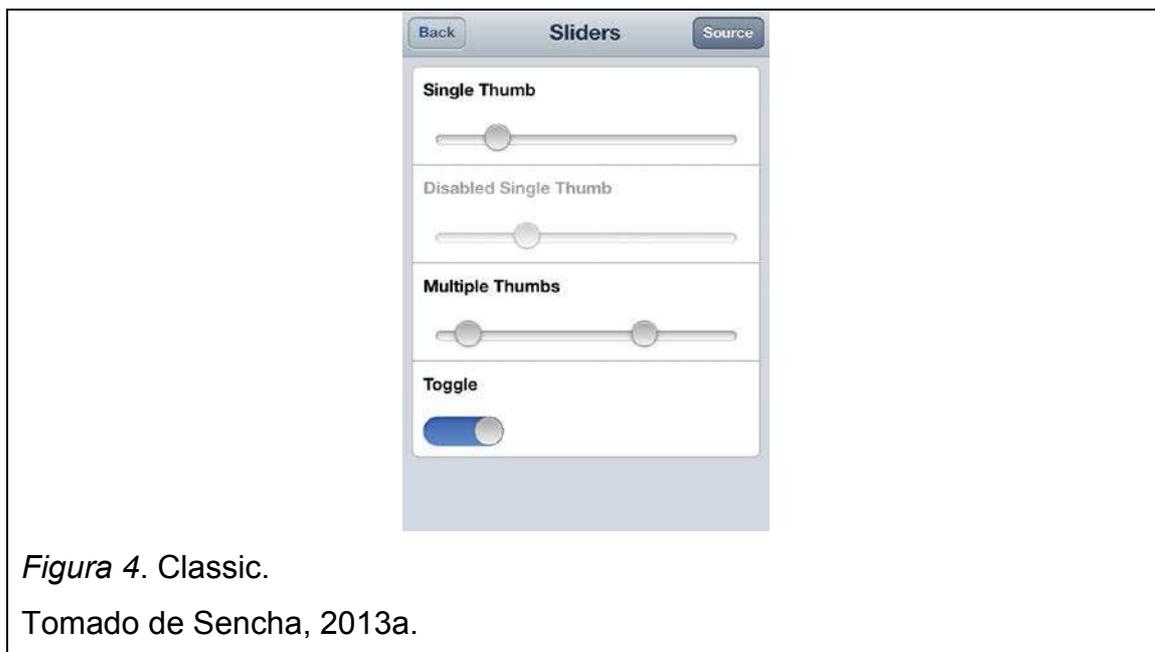
1.3.2.3. Sencha Touch

Según Sencha, 2013 Sencha Touch es una interfaz de usuario, una biblioteca de JavaScript, o Framework, construido específicamente para aplicaciones web móviles. El propósito principal de este framework es el desarrollo de interfaces de usuario para aplicaciones, utilizando herramientas y lenguajes web.

Permite a los desarrolladores construir aplicaciones que se vean y sientan como aplicaciones nativas para varios sistemas operativos con solo ciertos cambios entre ellas permitiendo reducir ampliamente el tiempo de desarrollo. Está totalmente basado en estándares web como HTML5, CSS3 y JavaScript.

Según datos sacados de su página web, desde su lanzamiento en el 2010 más de 500000 desarrolladores han descargado Sencha Touch y han creado miles de aplicaciones. Son utilizadas por alrededor del 50% de las Fortune 100 y es parte de 8 de las 10 instituciones financieras más grandes.

Sencha Touch provee temas que se aplican a cada sistema operativo para dar una experiencia de usuario que se acerque en gran manera a la experiencia que proveen las aplicaciones nativas de cada uno de los SO. Las figuras 4 y 5 que se muestran a continuación son un ejemplo de los temas que se puede usar para el desarrollo.



Por diseño de la aplicación, se mantendrá la misma interfaz en todas las plataformas para mantener una sensación unificada del producto que se desea desarrollar.

Sencha Touch permite a sus desarrolladores modificar el código fuente del mismo para la creación de aplicaciones que se ajusten a las necesidades de sus desarrolladores eliminando restricciones que podrían existir al utilizar un framework web. Se basa en el modelo vista controlador (MVC).

Características:

Sencha Touch incluye un conjunto de controles o componentes de interfaz gráfica de usuario (GUI) para su uso en aplicaciones web para dispositivos móviles, estos componentes se han optimizado para facilitar la entrada por medios táctiles como los dedos, algunos de los componentes son:

- Botones con temas y efectos específicos del dispositivo.
- Elementos de formulario, como campos de texto para el correo electrónico, selector de fechas, deslizadores, selectores y combo boxes.
- Listas que permitan desplazamiento con regulación de fuerza y una barra de índice.
- Barras de herramientas y menús.
- Pestañas móviles.
- Barra de herramientas inferior.

A todos los componentes se pueden aplicar temas de acuerdo con el dispositivo de destino. Esto se hace utilizando SASS, un lenguaje de estilos construido sobre CSS.

Comparación con aplicaciones nativas:

Las aplicaciones nativas en los dispositivos móviles principalmente difieren de las aplicaciones web por su acceso a los componentes del dispositivo, como la cámara, brújula, y el micrófono, pese a que Sencha Touch tiene cierto soporte

para esto, requiere de una herramienta que empaquete la aplicación en un entorno nativo y este es otro de los frameworks que hemos seleccionado para este proyecto llamado Córdova o Phonegap.

Sencha Touch se puede utilizar en conjunto con Córdova, con Phonegap o con su propio empaquetador nativo para el desarrollo de aplicaciones nativas orientadas a dispositivos móviles que tengan acceso a componentes que solo las aplicaciones nativas tienen posibilidad de acceder, la capacidad de estas aplicaciones se limitan a la API de los empaquetadores como es en este caso Phonegap, alternativamente en Sencha Touch se puede utilizar su propio empaquetador nativo.

Este framework debe utilizarse con una herramienta extra de la empresa Sencha llamado Sencha CMD.

Sencha Cmd es una herramienta multiplataforma que funciona en la línea de comandos del SO en el que se instale, proporciona tareas automatizadas que ayudan durante todo el ciclo de vida de las aplicaciones que se desarrollen con Sencha Touch desde la generación de un nuevo proyecto hasta la puesta en producción de la misma aplicación.

Sencha Cmd proporciona un conjunto de características que nos permiten ahorrar tiempo de desarrollo, las mismas que son:

- Herramientas de generación de código: Permiten generar aplicaciones completas y ampliar estas aplicaciones con nuevos componentes de MVC.
- Compilador de JS: Un compilador de JavaScript que entiende la semántica del framework Sencha. El compilador puede optimizar muchos procesos de la semántica de Sencha que nos permita reducir el tiempo de carga de las aplicaciones.
- Servidor Web: Proporciona un servidor web ligero que permite ver archivos desde localhost

- Empaquetador Nativo: Permite convertir una aplicación Sencha Touch en una aplicación nativa móvil que tenga acceso a la funcionalidad del dispositivo y puede ser distribuido en tiendas de aplicaciones
- Sistema de gestión de paquetes: para una fácil integración de paquetes creados por otros usuarios o del repositorio del paquete Sencha.
- Herramientas de afinación.
- Gestión de espacio de trabajo.
- Captura de imagen: Convierte CSS3 en sprites (colección de imágenes) para navegadores antiguos.
- Configuración flexible del Sistema.
- Registro robusto para ayudarle a entender el funcionamiento interno de los comandos y facilitar la solución de problemas.
- Software de terceros: Sencha Cmd incluye una versión de software requerido, Compass, Sass y Apache Ant.

1.3.2.4. Compass

Es una herramienta para autoría de CSS basada en SASS y de código abierto, permite programar en SASS y convierte este código en CSS minimizado y organizado para optimizar tiempos de respuesta y organización en la aplicación.

Sass es una extensión de CSS3 que añade reglas anidadas, variables, mixins, selectores de herencia, y más. SASS genera un CSS bien formado y hace que las hojas de estilo sean más fáciles de organizar y mantener. Eppstein, s.f.

1.3.2.5. Phonegap

Según Adobe. 2014a, PhoneGap es un marco de desarrollo móvil producido por Nitobi, adquirido por Adobe Systems, permite crear aplicaciones para

dispositivos móviles utilizando JavaScript, HTML5 y CSS3, en lugar de lenguajes nativos como Objective-C o Java.

Las aplicaciones resultantes son híbridas, es decir, no son ni verdaderamente nativas ni web ya que se utiliza fragmentos de código de ambos mundos y permiten utilizar, mediante APIs, el hardware del dispositivo como el GPS, cámara, acelerómetro, etc. para formar la aplicación.

A partir de la versión 1.9 en adelante es posible incluso mezclar libremente fragmentos de código nativo e híbrido.

Según (Adobe, 2013b). Phonegap está basado en Apache Córdova, que es un software de código abierto que maneja todas las APIs de conexión con el hardware del dispositivo. Es el motor de Phonegap.

El núcleo de las aplicaciones creadas con PhoneGap utiliza HTML5 y CSS3 en su interfaz gráfica, y JavaScript para programar la lógica de la aplicación, aunque HTML5, por convenciones con los fabricantes o arquitectura del dispositivo, proporciona acceso a ciertas partes del hardware de los dispositivos.

El soporte de HTML5 no es consistente a través de los navegadores móviles, sobretodo en versiones anteriores de Android. Para superar estas limitaciones, PhoneGap empaqueta el código HTML5 dentro de un WebView nativo en el dispositivo y utilizando una interfaz de funciones externas (FFI) permite acceder a los recursos nativos del dispositivo.

PhoneGap también es capaz de extender su funcionalidad con plugins nativos que permiten a los desarrolladores agregar capacidades extra a la cuales se puede acceder, como si fueran librerías o llamadas de JavaScript, lo que permite la comunicación directa entre la capa nativa, y la capa HTML5.

PhoneGap de paquete incluye plugins básicos que permiten el acceso al acelerómetro del dispositivo, cámara, micrófono, brújula, sistema de archivos y algunas más.

Sin embargo, las aplicaciones basadas en tecnologías web pueden tener menor rendimiento que las aplicaciones nativas, por lo que el manejo de memoria o cuidado en la programación es esencial para mantener un buen funcionamiento y sensación de aplicación nativa que se quiere obtener.

Phonegap actualmente soporta varios sistemas operativos o plataformas y el acceso al hardware de cada dispositivo puede variar.

La tabla siguiente muestra un cuadro comparativo de dispositivos soportados en Phonegap.

Tabla 1. Dispositivos soportados en OS populares.

	 iOS	 iOS	 Android	 OS 5.x	 OS 6.0+	 WebOS	 WP7	 Symbian	 Bada
	Phone 7, Phone 3G	Phone 3GS and newer	Android	OS 5.x	OS 6.0+	WebOS	WP7	Symbian	Bada
ACCELEROMETER	✓	✓	✓	✓	✓	✓	✓	✓	✓
CAMERA	✓	✓	✓	✓	✓	✓	✓	✓	✓
COMPASS	✗	✓	✓	✗	✗	✗	✓	✗	✓
CONTACTS	✓	✓	✓	✓	✓	✗	✓	✓	✓
FILE	✓	✓	✓	✓	✓	✗	✓	✗	✗
GEOLOCATION	✓	✓	✓	✓	✓	✓	✓	✓	✓
MEDIA	✓	✓	✓	✗	✗	✗	✓	✗	✗
NETWORK	✓	✓	✓	✓	✓	✓	✓	✓	✓
NOTIFICATION (ALERT)	✓	✓	✓	✓	✓	✓	✓	✓	✓
NOTIFICATION (SOUND)	✓	✓	✓	✓	✓	✓	✓	✓	✓
NOTIFICATION (VIBRATION)	✓	✓	✓	✓	✓	✓	✓	✓	✓
STORAGE	✓	✓	✓	✓	✓	✓	✓	✓	✗

Tomado de Intel, 2012.

1.3.3. Servidor y Bases de Datos

El servidor para el sistema que se desarrollará será alojado en los servidores de Amazon, por su confiabilidad, disponibilidad y escalabilidad entre otras cosas, debido a lo cual será implementado como un servidor virtual y con una interfaz Linux escogida por los desarrolladores, al igual que los servidores web que se necesitarán para este proyecto.

1.3.3.1. Amazon Linux AMI

“La AMI de Amazon Linux es una imagen de Linux mantenida y compatible que ofrece Amazon Web Services para su uso en Amazon Elastic Compute Cloud (Amazon EC2). Está diseñada para proporcionar un entorno de ejecución estable, seguro y de alto rendimiento para aplicaciones que se ejecuten en Amazon EC2. También incluye paquetes que permiten una fácil integración con AWS, incluidas herramientas de configuración de lanzamiento y muchas bibliotecas y herramientas populares de AWS.

Amazon Web Services también proporciona actualizaciones continuas de seguridad y mantenimiento para todas las instancias ejecutadas en la AMI de Amazon Linux. La AMI de Amazon Linux se proporciona sin cargo adicional a los usuarios de Amazon EC2.” Amazon, 2013d.

1.3.3.2. Amazon Elastic Compute Cloud (Amazon EC2)

“Amazon Elastic Compute Cloud (Amazon EC2) es un servicio web que proporciona capacidad informática con tamaño modificable en la nube. Está diseñado para facilitar a los desarrolladores recursos informáticos escalables basados en web.

La sencilla interfaz de servicios web de Amazon EC2 permite obtener y configurar su capacidad con una fracción mínima, proporciona un control completo sobre sus recursos informáticos y permite ejecutarse en el entorno

informático acreditado de Amazon. Amazon EC2 reduce el tiempo necesario para obtener y arrancar nuevas instancias de servidor en minutos, lo que permite escalar rápidamente la capacidad, ya sea aumentándola o reduciéndola, según cambien sus necesidades. Amazon EC2 cambia el modelo económico de los servidores de informática en la nube al permitirle pagar solo por la capacidad que utilice realmente. Amazon EC2 proporciona a los desarrolladores las herramientas necesarias para crear aplicaciones resistentes a errores y para aislarse de los casos de error más comunes.” Amazon, 2013a.

1.3.3.3. Amazon Relational Database Service (Amazon RDS)

“Amazon Relational Database Service (Amazon RDS) es un servicio web que facilita las tareas de configuración, operación y escalado de una base de datos relacional en la nube. Proporciona capacidad rentable y de tamaño modificable y, al mismo tiempo, gestiona las tediosas tareas de administración de la base de datos, lo que le permite centrarse en sus aplicaciones y en su negocio.

Amazon RDS le permite acceder a todas las funciones de un motor de base de datos MySQL, Oracle, Microsoft SQL Server o PostgreSQL conocido. Esto supone que el código, las aplicaciones y las herramientas que ya utiliza en la actualidad con sus bases de datos existentes funcionarán con Amazon RDS. Amazon RDS incluye parches automáticamente en el software de base de datos y realiza una copia de seguridad de la misma, almacenando las copias de seguridad durante un período de retención definido por el usuario y permitiendo la recuperación a un momento dado. Podrá beneficiarse de la flexibilidad que supone poder escalar los recursos informáticos o la capacidad de almacenamiento asociada con su instancia de base de datos por medio de una única llamada a API.” Amazon, 2013b.

1.3.3.4. Amazon Simple Storage Service (Amazon S3)

“Amazon S3 es almacenamiento para Internet. Está diseñado para facilitar a los desarrolladores recursos informáticos escalables basados en la Web.

Amazon S3 proporciona una sencilla interfaz de servicios web que puede utilizarse para almacenar y recuperar la cantidad de datos que desee, cuando desee y desde cualquier parte de la Web.

Concede acceso a todos los desarrolladores a la misma infraestructura económica, altamente escalable, fiable, segura y rápida que utiliza Amazon para tener en funcionamiento su propia red internacional de sitios web. Este servicio tiene como fin maximizar las ventajas del escalado y trasladar estas ventajas a los desarrolladores.” Amazon, 2013c.

1.3.3.5. Servidor HTTP Apache

Este es un proyecto de desarrollo de software colaborativo con el objetivo de crear un servidor HTTP lo suficientemente robusto para ser usado a nivel comercial con todas las características que esto implica, además de ser de código abierto. El proyecto es parte de la Fundación de Software Apache y es administrado por voluntarios y usuarios alrededor del mundo. Apache Software Foundation, 2014.

1.3.3.6. WSGI (Web Server Gateway Interface)

Es una especificación universal para la comunicación entre servidores web, servidores de aplicaciones y aplicaciones web que se usa para Python. WSGI.org., s.f.

1.3.3.7. MySQL

MySQL es uno de los gestores de base de datos relacionales de código abierto más populares en el mundo, su código se encuentra bajo licencia GNU (General Public License), lo cual ha hecho que existan millones de

instalaciones y que empresas de gran tamaño como Facebook, Google, Adobe, entre otras, lo usen para sus operaciones diarias debido a su gran robustez, a su amplia comunidad de apoyo y a su flexibilidad a la hora de gestionar cantidades considerables de datos. Es muy implementado en sitios web bajo la arquitectura LAMP (Linux, Apache, MySQL, Perl/PHP/Python).

Actualmente es desarrollado por Oracle.

1.3.3.8. Alternative PHP Cache (APC)

Es un marco de trabajo de código abierto que se almacena en caché la salida del compilador de PHP, reduciendo así el análisis y la sobrecarga de procesamiento y solicitudes de disco. En algunos casos esto puede acelerar hasta 3 veces la velocidad de generación de páginas. (The PHP Group, 2014). Ha sido usado por grandes portales como Facebook, el cual participó en el proyecto

2. Desarrollo del sistema

2.1. Visión

El proyecto está orientado a los actuales y futuros usuarios de las diferentes cadenas de cines del Ecuador que no cuentan con una herramienta de acceso a la información de las carteleras de los complejos, de una manera centralizada.

Actualmente cada complejo de cines publica la información de su cartelera de manera individual ya sea a través de medios impresos o digitales, lo que

ocasiona pérdida de tiempo en la búsqueda de películas en horarios y fechas deseadas.

El prototipo a desarrollar permitirá buscar carteleras, películas, horarios de los diferentes complejos de cines del Ecuador brindando varios filtros como: ciudad, cine, horas, etc. Además incluirá una simulación de compra de un boleto para cualquier película o cine.

Se generará una versión para web y otra para dispositivos móviles iOS y Android.

Adicionalmente, con el propósito de brindar un mayor sentido de realidad a la compra de boletos dentro del prototipo, se creó una simulación de la conexión con los sistemas de una cadena de cines mediante tablas de base de datos para verificar el inventario de los complejos, crear registro de las compras y generar reportes de las mismas. De esta manera se asegura la existencia de boletos al momento de la compra y se informa al usuario sobre la disponibilidad.

En el evento de que varias compras generasen una inconsistencia de inventario y por lo tanto un error en la compra, se guardará un registro de dichos errores para un proceso de devolución.

2.1.1. Beneficios y limitaciones.

En base a la visión, estos serán los beneficios y limitaciones que el sistema tendrá.

Experiencia de usuario: Hacer de la consulta de carteleras e información un proceso lo más rápido, interactivo e intuitivo posible.

- Ver información de una película como leer la sinopsis, duración, género, director y elenco. y ver el tráiler.
- Visualización de películas (tráiler, sinopsis, duración, género, elenco, director, censura) y horarios que se encuentren en cartelera en los siguientes periodos de tiempo:
 - Semana de la consulta.
 - Día de la consulta.
- Visualización de películas (tráiler, sinopsis, duración, género, elenco, director, censura) y horarios en cartelera de cada uno de los complejos de cines.
- Filtro de horario de funciones para el día de la consulta, el día siguiente o dos días después dentro de cada complejo o película.
- Visualización de funciones (película y complejo) dentro de las próximas dos horas.
- Visualización de salas de cine (teléfonos, dirección) y horarios de cada una.

Integración Social: Integrar el sistema con Facebook para incentivar su uso y darle un valor agregado.

- Compartir la información de la película (poster, nombre).
- Calificar y compartir la calificación de una película en un rango de 1 a 5 estrellas.
- Visualizar la calificación de una película basada en calificaciones anteriores hechas por usuarios del sistema.

- Guardar y visualizar los cines favoritos para un rápido acceso a los mismos.
- Guardar, compartir y visualizar los estrenos esperados.

Uso de Hardware: Mostrar contenido en base a la locación del usuario.

- Mostrar películas y cines dentro de la ciudad en la que el usuario se encuentre mediante el uso del GPS del dispositivo.
- Selección de la ciudad en caso de que el GPS no esté disponible guardando las preferencias de locación automáticamente.

Actualización de Información: Mantener el sistema actualizado con la funciones de la semana actual. Mediante el envío de información por parte de los cines se realizará el siguiente proceso para actualizar la base de datos:

- Ingreso manual a través de sitio de administración de nuevas películas en estreno juntos con su información (poster, elenco, sinopsis, etc)
- Normalizar archivo Excel mediante herramienta Google Refine
- Convertir archivo .xlsx a CSV (comma separated value) para ingresar en base de datos
- Correr script SQL para ingreso de archivo CSV a base de datos

2.1.2. Matriz de riesgos

Tabla 2. Tabla de la matriz de riesgos

#	Declaración de Riesgo		(Escala)	(Escala)	Exposición	Mitigación	Contingencia	Disparador	Asignado
	Condición	Consecuencia	Probabilidad	Impacto					
1	Si no existe colaboración del profesor guía	No se puede avanzar con las correcciones y retroalimentación	4	8	32	Estar en constante contacto con el profesor guía	Informar al coordinador de la carrera	Si no se logra contactar al profesor durante tres semanas	JG
2	Si un miembro del equipo no colabora con el desarrollo del proyecto	No se puede continuar con el desarrollo	5	9	45	Planificar el desarrollo semanalmente o mediante los sprints	Conversar con la persona para generar un cambio de actitud	Si no se tiene ningún avance de la persona en dos semanas	JG
3	Si Amazon Web Services deja de prestar sus servicios	Se tiene que cambiar de servicio de computación y alojamiento	1	8	8	Estar siempre pendiente del estado del servicio de Amazon	Buscar inmediatamente un nuevo servicio	Si el servicio no está disponible por más de 3 días	SM
4	Si el alcance no fue correctamente definido	El tiempo para completar el proyecto puede ser muy corto	4	7	28	Desarrollar de la manera más ágil posible	Solicitar una prórroga en el plazo de entrega	Si no se ha completado el 80% del proyecto faltando 2 meses para la entrega	JG
5	Si el sistema de compra de tickets es de alta complejidad	El tiempo de desarrollo se alarga demasiado	6	6	36	Buscar la plataforma más eficaz para implementar	Cambiar de plataforma	Si el desarrollo de esta funcionalidad toma más de un mes	SM
6	Si las cadenas de cine deciden interrumpir el envío semanal de información	No se puede actualizar la información de carteleras	5	9	45	Mantener una buena relación con los cines	Crear scripts de lectura de páginas web de cada cine	Si no se recibe la información semanal por más de dos semanas	SM
7	Si la experiencia y aptitudes de los desarrolladores no es adecuada	El desarrollo del proyecto no se va a dar de manera ágil	4	8	32	Capacitación e investigación constante	Pedir ayuda a profesor guía	Si los sprints se retrasan repetidamente	JG

2.1.3. Cronograma General

Tomando en cuenta las habilidades y experiencia en conjunto del equipo de trabajo al hacer aplicaciones o sitios web se ha estimado de manera empírica siguiendo las mejores prácticas de SCRUM los tiempos necesarios para completar las tareas que se han propuesto en este proyecto. Estos tiempos tomados en semanas de trabajo se pueden ver en la tabla 3.

Tabla 3. Cronograma general del proyecto

	1ra Semana	2da Semana	3ra Semana	4ta Semana	5ta Semana	6ta Semana	7ma Semana	8va Semana	9na Semana	10ma Semana	11va Semana	12va Semana	13va Semana	14va Semana	15va Semana	16va Semana	17va Semana	18va Semana	Total
Visionamiento																			2 semanas
Planeación																			3 semanas
Construcción																			16 semanas
Pruebas																			5 semanas
Implementación																			3 semanas

2.1.4. Historias de Usuario

1. Ingreso al sistema usuario social

Como usuario social del sistema,

Quiero ingresar al sistema,

Para consultar carteleras de cine de la ciudad en la que me encuentre.

Criterios de aceptación:

Se deben ingresar las credenciales de Facebook.

Al ingresar un usuario de Facebook que no existe no permitirá ingresar a Facebook.

Se debe autorizar a la aplicación a usar información de Facebook.

Si no se autoriza usar información a Facebook, no debe ingresar a Facebook.

En caso de no disponer de GPS o de no autorizar el uso del mismo, permitir selección manual de ciudad.

2. Ingreso al sistema usuario anónimo

Como usuario anónimo del sistema,

Quiero ingresar al sistema,

Para consultar carteleras de cine de la ciudad en la que me encuentre.

Criterios de aceptación:

Se debe desplegar las carteleras de las películas con las restricciones al usuario anónimo correspondientes.

En caso de no disponer de GPS o de no autorizar el uso del mismo, permitir selección manual de ciudad.

3. Consulta de cartelera por película en el día actual de la ciudad

Como usuario,

Quiero ver las películas de todos los cines de la ciudad del día actual,

Para consultar los horarios.

Criterios de aceptación:

Se deben ver únicamente horarios de las películas del día actual en la ciudad seleccionada.

4. Consulta de cartelera por película del día siguiente

Como usuario,

Quiero ver las películas de todos los cines de la ciudad del día siguiente a la consulta,

Para consultar los horarios.

Criterios de aceptación:

Se deben ver únicamente horarios de las películas del día siguiente en la ciudad seleccionada.

5. Consulta de cartelera por película de pasado mañana

Como usuario,

Quiero ver las películas de todos los cines de la ciudad de pasado mañana,

Para consultar los horarios.

Criterios de aceptación:

Se deben ver únicamente horarios de las películas del día después de mañana en la ciudad seleccionada.

6. Consulta de cartelera por película de la semana actual

Como usuario,

Quiero ver las películas de todos los cines de la ciudad de la semana actual,

Para consultar los horarios.

Criterios de aceptación:

Se deben ver únicamente horarios de las películas de la semana actual en la ciudad seleccionada.

7. Consulta de cartelera por película de las próximas dos horas

Como usuario,

Quiero ver las películas de todos los cines de la ciudad en las próximas dos horas,

Para consultar los horarios.

Criterios de aceptación:

Se deben ver únicamente horarios de las películas dentro de las próximas dos horas en la ciudad seleccionada.

8. Consulta de próximos estrenos

Como usuario,

Quiero ver todos los próximos estrenos en los cines de la ciudad,

Para consultar la información general (tráiler, poster, sinopsis, director, elenco, duración, país de origen, género, censura) de la película.

Criterios de aceptación:

Se deben ver únicamente películas que estén próximas a estrenarse y que todavía no tengan funciones en ninguna categoría de horarios.

Toda la información (tráiler, poster, sinopsis, director, elenco, duración, país de origen, género, censura) debe corresponder a la película seleccionada.

El tráiler debe ajustarse a la pantalla del dispositivo.

9. Consulta de cartelera por cine del día actual

Como usuario,

Quiero ver las películas del cine seleccionado en el día actual,

Para consultar los horarios.

Criterios de aceptación:

Se permite seleccionar el complejo de cines.

Se permite modificar el complejo de cines.

Se deben desplegar únicamente horarios de las películas del día actual en el cine seleccionado.

10. Consulta de cartelera por cine del día siguiente

Como usuario,

Quiero ver las películas del cine seleccionado del día siguiente,

Para consultar los horarios.

Criterios de aceptación:

Se deben ver únicamente horarios de las películas del día siguiente en el cine seleccionado.

11. Consulta de cartelera por cine de pasado mañana

Como usuario,

Quiero ver las películas del cine seleccionado de pasado mañana,

Para consultar los horarios.

Criterios de aceptación:

Se deben ver únicamente horarios de las películas de pasado mañana en el cine seleccionado.

12. Consulta de información general de una película

Como usuario,

Quiero consultar la información general (tráiler, poster, sinopsis, director, elenco, duración, país de origen, género, censura) de la película seleccionada,

Para tener información de interés sobre la película.

Criterios de aceptación:

Toda la información (tráiler, poster, sinopsis, director, elenco, duración, país de origen, género, censura) debe corresponder a la película seleccionada.

El tráiler debe ajustarse a la pantalla del dispositivo.

13. Consulta de información general de complejo

Como usuario,

Quiero consultar la información general (nombre, dirección, teléfono) de todos los complejos de cine de la ciudad,

Para tener información de interés sobre el cine.

Criterios de aceptación:

Se debe desplegar el teléfono y la dirección correspondiente al cine seleccionado.

14. Calificar película

Como usuario social,

Quiero calificar una película en una escala de 1 a 5 estrellas y compartirla en mi muro de Facebook,

Para expresar mis gustos de cine con mis amigos.

Criterios de aceptación:

Si el usuario es anónimo no podrá calificar la película.

Comprobar que la calificación se publicó en Facebook.

Comprobar que la calificación se registró en la base de datos.

15. Guardar película a lista de estrenos esperados

Como usuario social,

Quiero guardar un próximo estreno en mi lista de estrenos esperados y compartirlo en mi muro de Facebook,

Para saber que estrenos quiero ver luego.

Criterios de aceptación:

Si el usuario es anónimo no podrá guardar el próximo estreno.

Comprobar que el próximo estreno se publicó en Facebook.

Comprobar que el próximo estreno se guardó en la lista de estrenos esperados.

Comprobar que el próximo estreno correcto del usuario correspondiente se registró en la base de datos.

16. Guardar complejo en lista de favoritos

Como usuario social,

Quiero guardar un complejo de cine en mi lista de cines favoritos,

Para consultar su información y cartelera más rápidamente.

Criterios de aceptación:

Si el usuario es anónimo no podrá guardar el cine como favorito.

Comprobar que el cine correcto se guardó en la lista de cines favoritos del usuario correspondiente.

Comprobar que el cine correcto se guardó como favorito para el usuario correspondiente en la base de datos.

17. Consultar películas calificadas

Como usuario social,

Quiero consultar la lista de mis películas calificadas,

Para saber cómo calificué a las películas.

Criterios de aceptación:

Si el usuario es anónimo, no tendrá lista de películas calificadas.

Deberán verse únicamente las películas que el usuario calificó previamente junto con su respectiva calificación.

18. Compra de boleto de función

Como usuario social,

Quiero consultar los horarios de funciones en los diferentes cines de la ciudad,

Para comprar boletos de una función.

Criterios de aceptación:

Se debe poder elegir la función y cantidad de entradas deseadas.

Se debe desplegar el detalle de la compra.

Se debe recibir una notificación de confirmación de la compra.

2.2. Planeación

2.2.1 Equipo de trabajo

El equipo de trabajo está compuesto por dos integrantes que han asumido los siguientes roles:

Javier Granda:

- Experiencia de usuario
- Administrador de Producto
- Pruebas
- Desarrollo

Sebastián Mansfield:

- Administrador del Proyecto
- Administrador de liberaciones.
- Desarrollo
- Prueba

2.2.1. Lista de Producto

La planificación del proyecto se basó en las historias de usuario, estas se tradujeron a ítems de la lista de producto con una importancia relativa asignada por el administrador del producto.

Tabla 4. Tabla de la lista de producto

ID	Nombre	Importancia	Estimación	Como Probar	Notas
1	Ingreso al sistema usuario social.	60	8	<ul style="list-style-type: none"> Ingresar al sistema. Poner credenciales de Facebook. Ingresar a base datos y verificar que el usuario se ha creado. 	Se debe autorizar a la aplicación a usar información de Facebook
2	Ingreso al sistema usuario anónimo.	40	2	<ul style="list-style-type: none"> Ingresar al sistema sin necesidad de ingresar credenciales. 	Se limitara algunas funcionalidades destinadas a usuarios con cuentas
3	Consulta de cartelera por película en el día actual.	90	8	<ul style="list-style-type: none"> Ingresar al sistema. Consultar si todas las películas que estén dando ese día, en todos los cines de la ciudad se encuentren en la lista. 	Buscará películas de todos los cines de la ciudad en la B.B. D.D
4	Consulta de cartelera por película del día siguiente.	70	2	<ul style="list-style-type: none"> Ingresar al sistema Consultar si todas las películas que estén dando al día siguiente, en todos los cines de la ciudad se encuentren en la lista. 	Buscará películas de todos los cines de la ciudad en la B.B. D.D
5	Consulta de cartelera por película del día después de mañana.	40	2	<ul style="list-style-type: none"> Ingresar al sistema. Consultar si todas las películas que estén dando 2 días después en todos los cines de la ciudad se encuentren en la lista. 	Buscará películas de todos los cines de la ciudad en la B.B. D.D
6	Consulta de cartelera por película de la semana actual.	80	6	<ul style="list-style-type: none"> Ingresar al sistema. Consultar si todas las películas que vayan o estén dando durante la semana, en todos los cines de la ciudad se encuentren en la lista. 	Buscará películas de todos los cines de la ciudad en la B.B. D.D
7	Consulta de cartelera por película de las próximas dos horas.	40	8	<ul style="list-style-type: none"> Ingresar al sistema. Consultar si todas las películas que estén dando en las próximas dos horas, en los cines de la ciudad se encuentren en la lista. 	Buscará películas de todos los cines de la ciudad en la B.B. D.D
8	Consulta de próximos estrenos.	50	8	<ul style="list-style-type: none"> Ingresar al sistema. Consultar que las películas que aparezcan ahí no estén en cartelera todavía. 	Se ingresara la película con un campo adicional en la B.B.D.D que indique si es estreno próximo. La opción guardar solo aparecerá si el usuario está conectado a Facebook
9	Consulta de cartelera por cine del día actual.	90	8	<ul style="list-style-type: none"> Ingresar al sistema Consultar en la página del cine si todas las películas que estén dando ese día, en ese cine en particular, se encuentren en la lista. 	Se debe consultar todas las películas del complejo, que estén dado exclusivamente ese día
10	Consulta de cartelera por cine del día siguiente.	70	2	<ul style="list-style-type: none"> Ingresar al sistema Consultar en la página del cine si todas las películas que estén dando al día siguiente, en ese cine en particular, se encuentren en la lista. 	Se debe consultar todas las películas del complejo, que estén dado exclusivamente el día siguiente
11	Consulta de cartelera por cine de dos días después	40	2	<ul style="list-style-type: none"> Ingresar al sistema Consultar en la página del cine si todas las películas que estén dando 2 días después, en ese cine en particular, se encuentren en la lista. 	Se debe consultar todas las películas del complejo, que estén dado exclusivamente dos días después

1 2	Consulta de información general de película	90	10	<ul style="list-style-type: none"> • Ingresar a película deseada • Consultar si toda la información desplegada es la correcta y no falta información. • Comprobar tráiler 	Se desplegara toda la información de la película que la base de datos tenga almacenada incluyendo un tráiler almacenado en YouTube
1 3	Consulta de información general de complejo	90	6	<ul style="list-style-type: none"> • Ingresar a cine deseado • Consultar si toda la información desplegada es la correcta. 	Se desplegara toda la información del cine que la base de datos tenga almacenada incluirá un logo del mismo.
1 4	Calificar película.	50	8	<ul style="list-style-type: none"> • Seleccionar calificación deseada • Guardar, Revisar si la calificación se encuentra guardada en la base de datos 	La calificación será de 1 a 5 y no almacenará decimales
1 5	Guardar película a lista de estrenos esperados	20	4	<ul style="list-style-type: none"> • Seleccionar estreno esperado. • Guardar. • Revisar si película se guardó correctamente en B.D.D 	Se almacenara solo el id de la película en la B.D.D
1 6	Guardar complejo en lista de favoritos	30	4	<ul style="list-style-type: none"> • Seleccionar cine deseado • Seleccionar guardar como favoritos. • Revisar si el cine se encuentra en lista de cines favoritos. 	Se almacenara solo el id del cine en la B.D.D
1 7	Consulta de películas calificadas	30	4	<ul style="list-style-type: none"> • Ingresar a página usuario • Seleccionar mis calificaciones • Revisar si todas las calificaciones realizadas por el usuario se encuentran ahí. 	Se consultara en la B.D.D todas las películas con las calificaciones hechas por el usuario.
1 8	Consulta lista de estrenos esperados	30	4	<ul style="list-style-type: none"> • Ingresar a página usuario • Seleccionar "Por Estrenar" • Revisar si todas las películas guardadas por el usuario se encuentran ahí. 	
1 9	Consulta lista de complejos de cine guardados	30	4	<ul style="list-style-type: none"> • Ingresar a página usuario • Seleccionar "Mis Complejos" • Revisar si todos los complejos guardados por el usuario se encuentran ahí. <p>2da Forma:</p> <ul style="list-style-type: none"> • Ingresar a lista de complejos de cines • Revisar en la sección de favoritos los complejos guardados por el usuario 	
2 0	Comprar Boletó	60	10	<ul style="list-style-type: none"> • Seleccionar película y horario deseado y seleccionar comprar. • Realizar el proceso de compra con el detalle de los boletos. • Revisar si la transacción se completó en la base de datos y sistema de pagos. 	Se usara un sistema de pagos ya existente.

2.3. Desarrollo

2.3.1. Criterios de Diseño

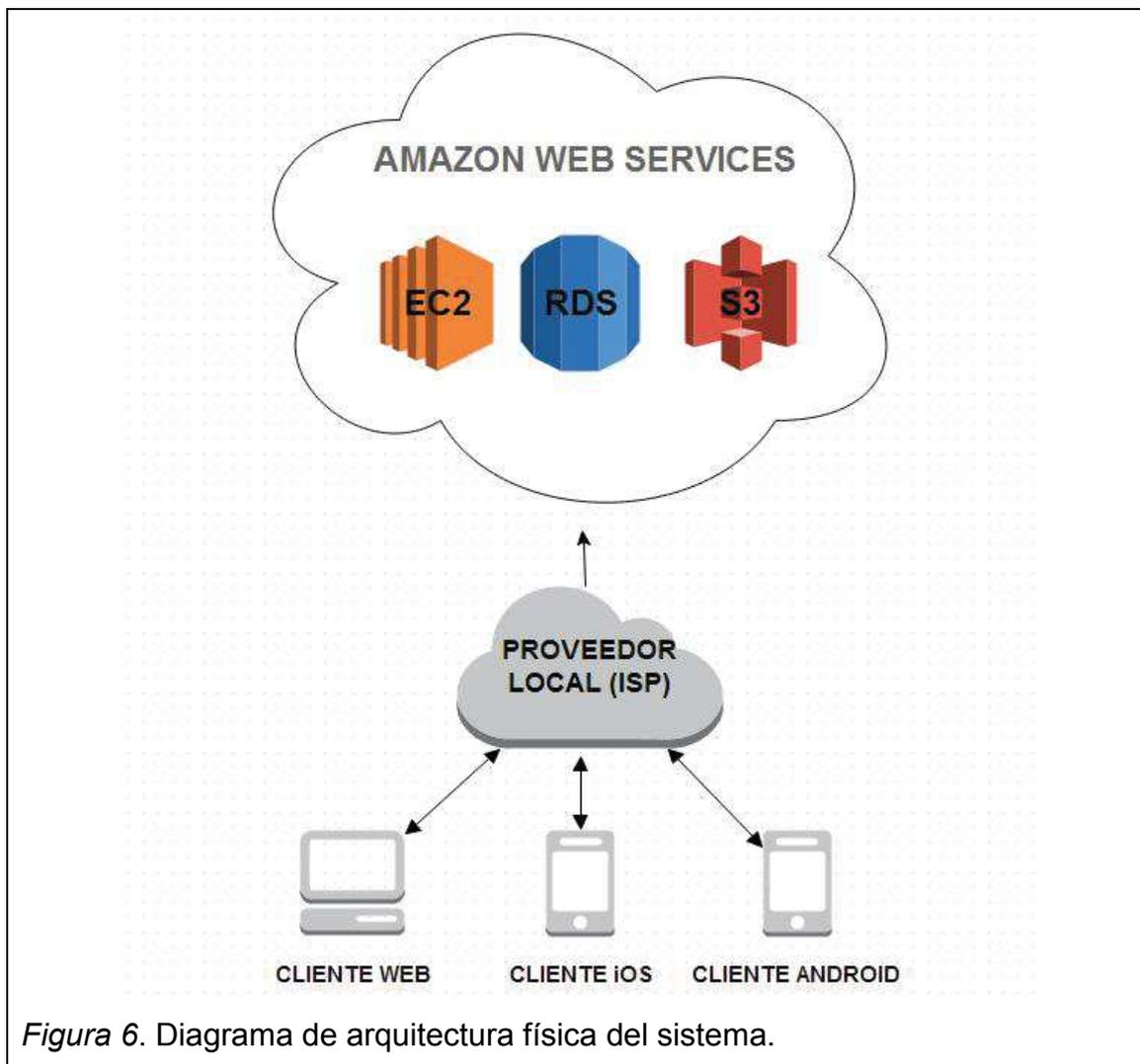
El diseño del sistema consta de dos ejes principales, uno orientado a los dispositivos móviles y otro orientado a la web clásica, por lo tanto se han seleccionado herramientas y metodologías que cumplan los objetivos de los siguientes criterios de diseño:

- **Alta disponibilidad de la información:** Debido a la naturaleza de los dispositivos móviles y a la alta demanda por parte de los usuarios hoy en día, es vital asegurar que la información se encuentre disponible lo más rápida posible y en cualquier momento.
- **Escalabilidad:** Debido a que la industria del cine en el Ecuador y el mundo se encuentra en constante crecimiento, se debe tomar en cuenta la habilidad con la que el sistema pueda crecer para satisfacer esta demanda y crecimiento.
- **Facilidad de administración:** Para asegurar la alta disponibilidad es indispensable manejar un sistema de administración de los datos eficiente y confiable.
- **Centralización de los datos:** Debido a que el sistema se presenta en múltiples plataformas y formatos, es vital centralizar el repositorio de información de la cual se alimentarán estas plataformas.
- **Experiencia de uso:** Desde el punto de vista del usuario, se debe considerar la facilidad y eficiencia con la que el sistema permitirá acceder e interactuar con la información disponible.

Basándonos en los casos de uso, el sistema a ser desarrollado centra sus beneficios en el usuario final, pero a la vez proporciona beneficios para el

equipo de desarrollo del producto. El usuario final, el cual utilizará cualquiera de las aplicaciones (IOS y Android) y el sitio web se beneficia de las funcionalidades de este software, estas incluyen limitaciones. A continuación se muestra el diagrama de la infraestructura física del sistema.

2.3.2. Infraestructura física



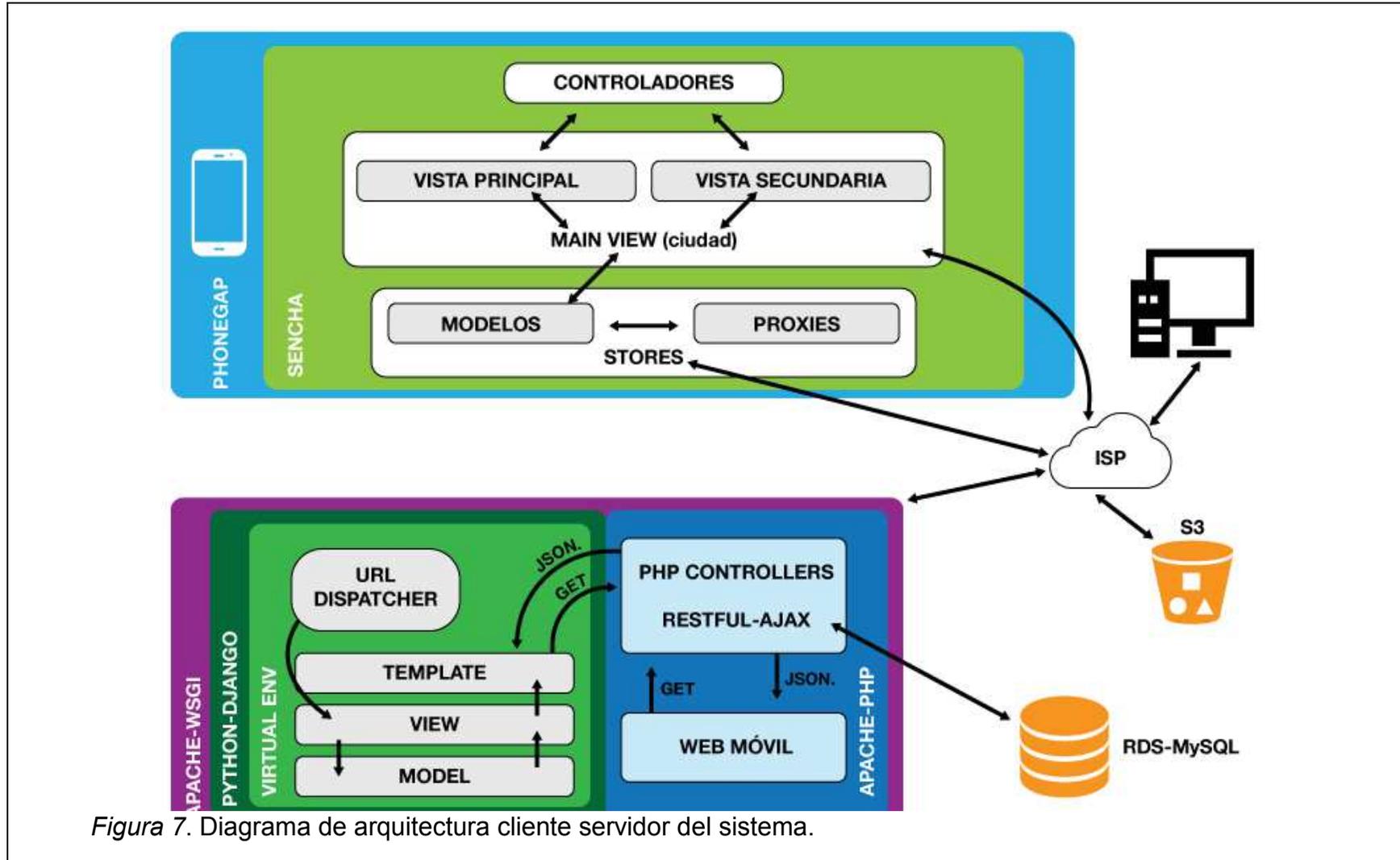
El sistema será alojado en un servidor, tanto para la parte web como para la parte móvil. Este debe permitir instalar varios software como fuese necesario, que ayuden al óptimo funcionamiento del sistema.

Para este propósito se usará el sistema EC2 de Amazon Web Services que nos permite instalar una instancia de Linux y configurarla como se desee.

Para almacenar imágenes que el sistema requiera, como serán posters o logos se utilizara S3 de Amazon, el cual permite almacenar una gran cantidad de imágenes y accederlas públicamente.

La base de datos utilizará el servicio RDS de Amazon. El cual gestiona de manera fácil la base de datos del sistema manteniendo respaldos y actualizando el servidor de manera automática.

2.3.3. Diagrama de arquitectura física y lógica



2.3.4. Modelo EER

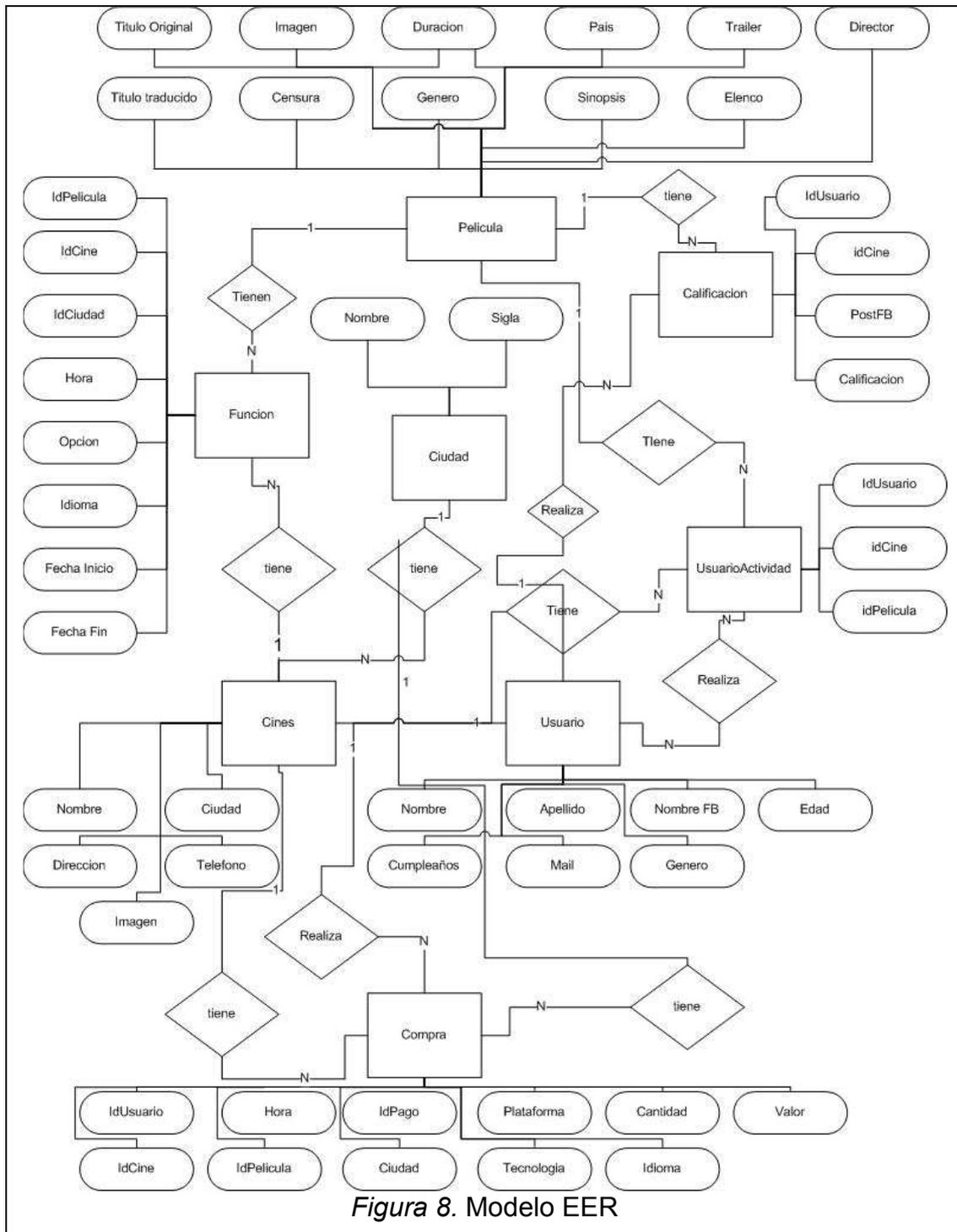


Figura 8. Modelo EER

2.3.5. Modelo físico de base de datos

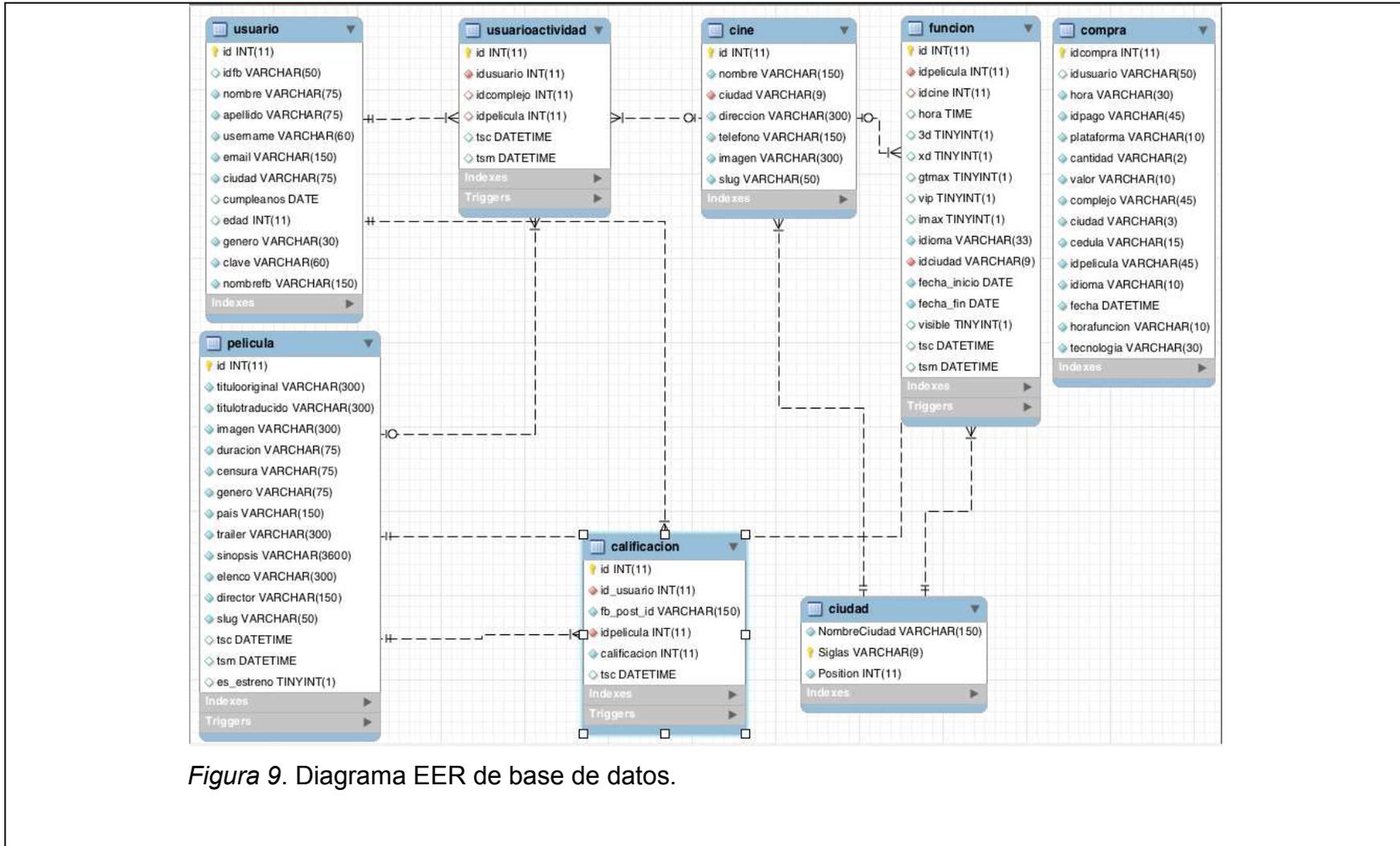
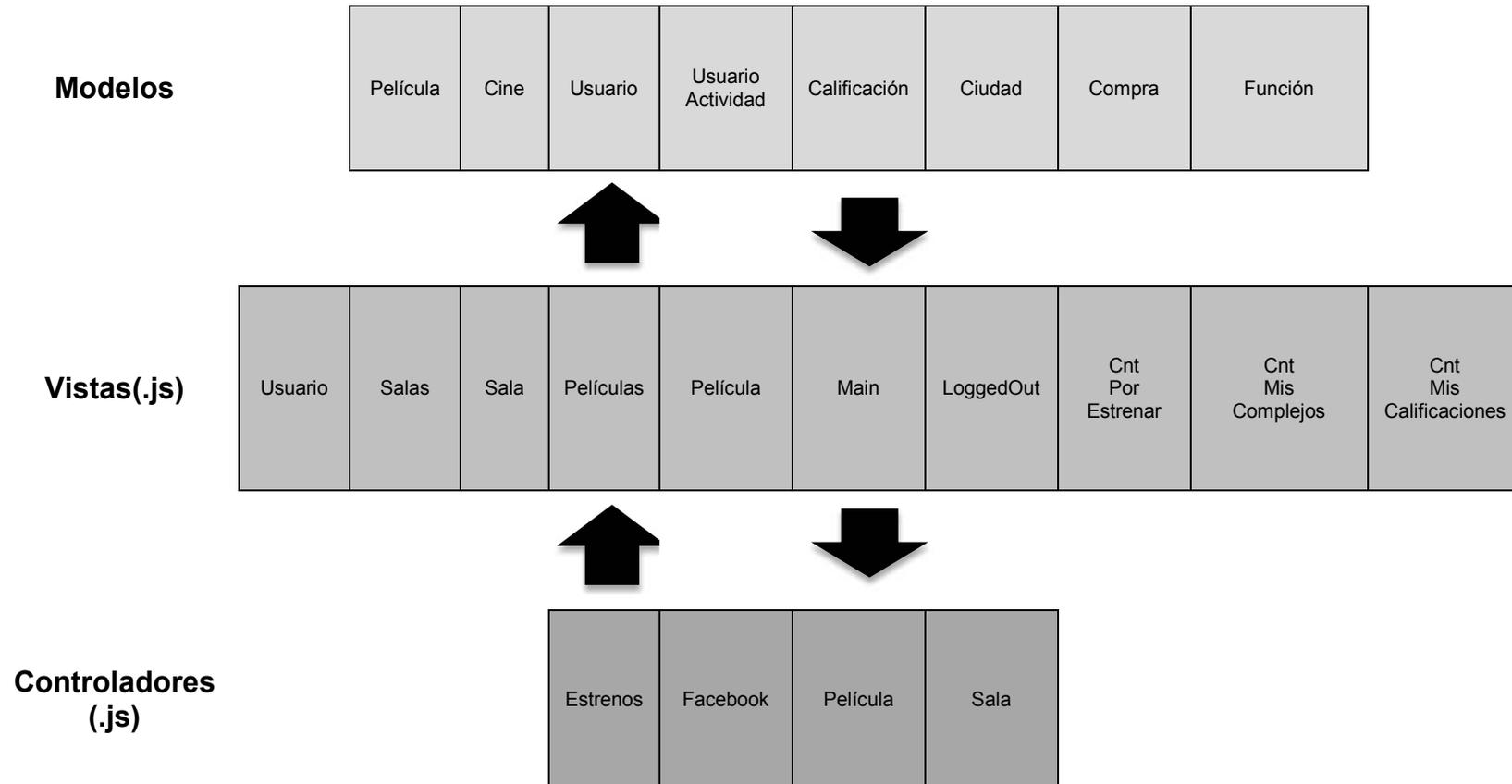


Figura 9. Diagrama EER de base de datos.

2.3.6. Tabla inventario arquitectura MVC (Modelo-Vista-Controlador)

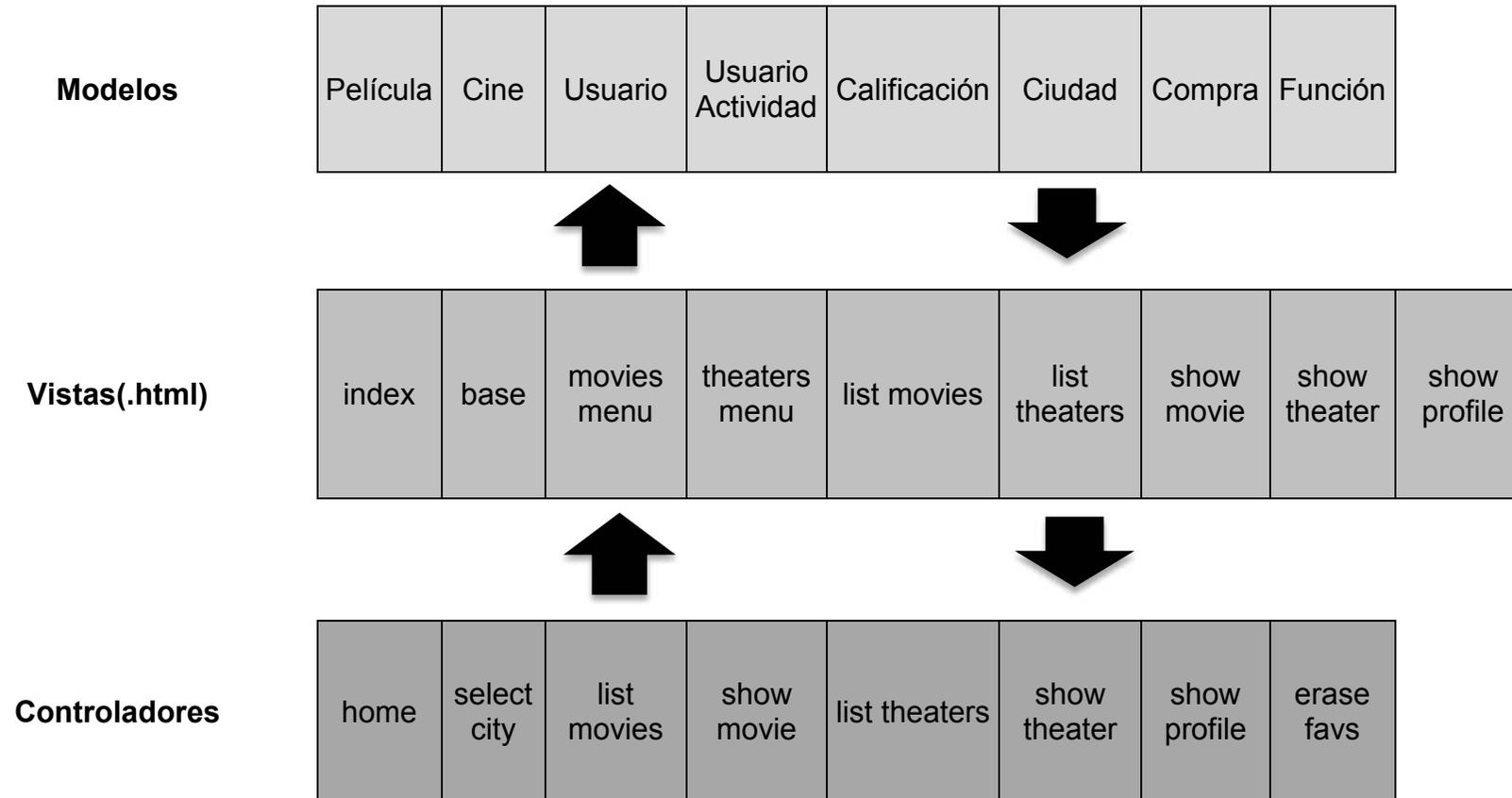
2.3.6.1. Aplicación

Tabla 6. Arquitectura MVC de aplicación móvil



2.3.6.2. Web

Tabla 7. Arquitectura MVC de sitio web



2.3.7. Funcionalidad

El ingreso al sistema se realizará de dos maneras con respecto a la locación.

Si el dispositivo cuenta con GPS y el usuario autoriza que la aplicación use este sistema, la aplicación buscará mediante el GPS la ciudad más cercana para desplegar los horarios y cines de la misma.

En el caso de no contar con GPS o si el usuario no autoriza su uso, se desplegará un menú para la selección manual de la ciudad.

La funcionalidad propuesta para el sistema se ha separado en módulos, los mismo que son:

2.3.7.1. Módulo Películas

Acceso sencillo y rápido a información de películas que están en cartelera, estas incluyen:

- Video promocional o tráiler.
- Póster promocional a escala en forma de thumbnail referencial.
- Título original y título traducido en caso de que existiese.
- Elenco donde consten únicamente los protagonistas.
- Director o directores principales.
- Duración en horas y minutos.
- Género de la película.
- Sinopsis la cual será una versión resumida para su fácil lectura por parte del usuario final.

En la página principal de la aplicación móvil se mostrarán cuatro carruseles desplazables y en el sitio web una cuadrícula navegable que contendrán la siguiente información:

Hoy en cartelera:

Visualizar poster y título de las películas, como ítems individuales, que estén en cartelera exclusivamente el día en el cual se consulte la aplicación. Esto permitirá acceder a la información anteriormente definida mediante la selección de uno de estos ítems.

Además de estar en los carruseles de la aplicación móvil, las secciones a continuación serán accesibles en el sitio web mediante un enlace en la página principal.

Próximas dos horas:

Visualizar poster, título, horario y complejo de las películas, como ítems individuales, que estén en cartelera durante las próximas dos horas a partir del momento de consulta de la aplicación.

En cartelera esta semana:

Visualizar poster y título de las películas, como ítems individuales, que se encuentren en cartelera durante la semana en la cual se consulte la aplicación.

Próximos estrenos:

Visualizar poster y título de las películas, como ítems individuales, que estén próximas a estrenar en cualquier complejo. Al acceder en alguno de estos ítems existirá un botón para guardar la misma en la opción “Mis estrenos esperados”, esta funcionalidad será definida más adelante.

2.3.7.2. Módulo Cines

La página principal de la aplicación móvil presentará dos secciones principales, favoritos y todos, que serán desplegadas en forma de listas. En el sitio web estas secciones serán accesibles mediante enlaces. Las listas

presentarán toda la información de los cines, la que incluye nombre del complejo, logo, dirección y teléfono principal. Las secciones son:

Favoritos: Representan los complejos que el usuario final ha seleccionado para su acceso rápido como sus complejos favoritos.

Todos: Representan todos los complejos de cine de la ciudad previamente seleccionada, o en la que se encuentra el usuario.

Al acceder a alguno de estos complejos se desplegará una nueva pantalla que contendrá la información básica del establecimiento, un botón para guardar como favorito y los horarios o funciones.

2.3.7.3. Módulo Perfil

En esta sección tendremos tres partes principales tanto para las aplicaciones móviles como para el sitio web:

- **Mis calificaciones:** Se podrá observar en forma de lista todas las calificaciones hechas por el usuario junto con información de la película calificada; consta del poster a escala, los nombres y la calificación.
- **Mis complejos:** Se podrá ver en forma de una lista los complejos que han sido guardados por el usuario, al seleccionarlos tendremos la opción de borrarlos de la lista y a su vez de la lista de rápido acceso.
- **Mis estrenos esperados:** En una lista se desplegará el poster y nombres de la película y se podrá acceder a las mismas para ver toda la información de la película de la forma explicada previamente.

2.3.7.4. Horarios

Esta no es una sección del sistema como tal pero estará integrada en las dos primeras secciones principales (Películas y Cines).

Al ingresar en la película, después de la información básica se desplegarán los horarios de todos los cines que se encuentren proyectando esta película en el horario seleccionado. En cada ítem de la lista de horarios se desplegará información básica del cine con su logo y a continuación los horarios. Existen tres filtros para visualizar los horarios, estos son:

- Hoy.
- Mañana.
- Pasado mañana.

Adicionalmente dentro de estos se desplegarán los formatos en los que se proyecta cada película, estos son:

- Lenguaje: subtitulada y español.
- Tipo de sala (GT-MAX, XD o normal).
- Tecnología de proyección (3D o 2D).
- Exclusividad (VIP o normal).

Adicionalmente, al ingresar a un complejo se desplegará la información, en el mismo formato, de los respectivos horarios de las películas que en este establecimiento se estén proyectando.

2.3.7.5. Compra de boletos

Dentro del sistema se implementará, a modo de prototipo, un sistema de compra de boletos que estará disponible al seleccionar cualquier función que se muestre dentro de la consulta por película o por complejo de cine.

Para el procesamiento del pago se utilizará la plataforma PayPal que brinda un ambiente SandBox (ambiente de pruebas) con el cual se pueden realizar pagos simulados.

El proceso para el usuario contará con los siguientes pasos:

1. Selección de función deseada
2. Selección de cantidad de entradas deseadas
3. Ingreso de número de cédula para propósitos de registro
4. Ingreso o creación de cuenta de PayPal con tarjeta de crédito asociada o escaneo de tarjeta de crédito en dispositivos móviles.
5. Confirmación de la compra con el detalle de la misma
6. Notificación de compra exitosa o de problemas en la tarjeta o el proceso de pago.

Mediante un servicio web se generará un reporte automático de transacciones realizadas en cada uno de los complejos de cine. Este reporte pretende brindar una mejor integración entre las cadenas de cine y el sistema propuesto.

2.3.8. Sprint Cero

2.3.8.1. Objetivo

Este sprint consiste únicamente de tareas de configuración, preparación e instalación de ambientes de trabajo necesarios para el desarrollo del sistema como tal, e incluye requisitos no funcionales por lo que solo se realiza la lista de sprint para mantener un registro de tiempo.

2.3.8.2. Alcance

Este sprint tendrá una duración de una semana al final de la cual se habrá configurado los ambientes de trabajo necesarios para empezar el desarrollo del prototipo.

2.3.8.3. Lista de Sprint

Tabla 8. Tabla de sprint 0

Tarea	Responsable	Plataforma	Estimado (h)	Real (h)	Estado
Configuración de servidor local Django de pruebas	JG	Web	4	4	Hecho
Configuración de servidor MAMP local de pruebas	JG	Web	4	4	Hecho
Creación de BDD, tablas y campos necesarios según análisis	Equipo	Web/Móvil	8	8	Hecho
Registro en AWS	SM	Web/Móvil	3	3	Hecho
Selección y activación de servicios AWS (RDS, EC2, S3)	SM	Web/Móvil	2	2	Hecho
Instalación de Actualizaciones de Linux, Ajuste de zonas horarias	JG	Web/Móvil	2	2	Hecho
Instalación de PHP y extensiones	SM	Web/Móvil	3	3	Hecho
Instalación de Apache Httpd	JG	Web/Móvil	2	2	Hecho
Instalación de MySQL	SM	Web/Móvil	3	3	Hecho
Instalación de Modulo PHP APC	SM	Web/Móvil	3	3	Hecho
Instalación de Modulo Mod-WSGI	JG	Web/Móvil	7	10	Hecho
Iniciar servicios Apache en el servidor	JG	Web/Móvil	1	1	Hecho
Tareas no planificadas	JG	Web	3	3	Hecho

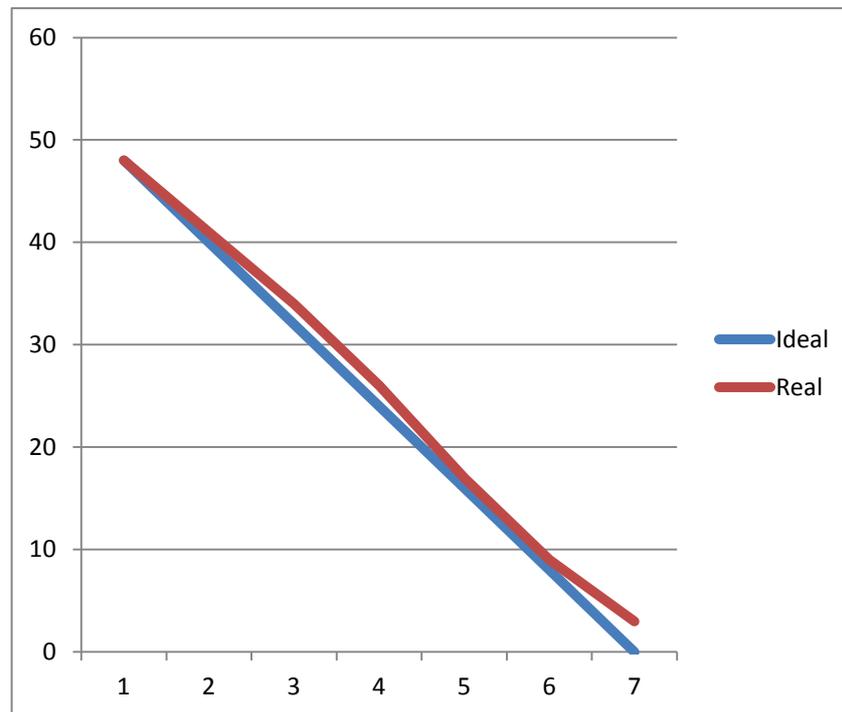


Figura 10. Gráfico Burndown del sprint 0.

2.3.8.4. Cierre y retrospectiva

Debido a que se trata de un sprint para satisfacer requisitos no funcionales, no forma parte del desarrollo como tal, sin embargo, hubieron retrasos dados por la falta de experiencia en el manejo de las herramientas. Fue muy importante las configuraciones realizadas para que los sprints de desarrollo se den de la manera más eficiente sin retrasos por errores de ambientes de desarrollo.

2.3.9. Primer Sprint

2.3.9.1. Objetivo

Implementar la consulta de cartelera de una película del presente día, del día siguiente, de pasado mañana, de la semana actual y de las próximas dos horas.

2.3.9.2. Alcance

Este sprint tendrá una duración de tres semanas al final de las cuales se deberá poder consultar, mediante un demo funcional, las carteleras de todos los cines en una ciudad de una película determinada y para todos los periodos de tiempo establecidos.

Acrónimos de desarrolladores: JG = Javier Granda; SM = Sebastián Mansfield.

Velocidad estimada: (suma de las horas de trabajo reales * factor de dedicación).

$$(120 * 75\%) = 90 \text{ horas}$$

2.3.9.3. Lista de Sprint

Tabla 9. Lista del primer sprint

User story	Tarea		Responsable	Plataforma	Estimado (h)	Real (h)	Estado
3, 4, 5 6, 7	Análisis y creación de funciones para categorizar horarios y filtrar periodos de tiempo				14	15	Hecho
	Subtarea	Función para categorizar la función por idioma y tipo (Español - Subtitulada, VIP, GT-MAX, XD, 3D)	JG	Web	5	5	Hecho
	Subtarea	Función para categorizar la función por idioma y tipo (Español - Subtitulada, VIP, GT-MAX, XD, 3D)	SM	Móvil	5	5	Hecho
	Subtarea	Función para devolver el valor numérico de tiempo a partir de un string	JG	Web	4	5	Hecho
3, 4, 5 6, 7, 8	Creación de archivos y funciones query de llamado y consulta a BDD				20	24	Hecho
3	Subtarea	Archivo PHP para devolver los horarios de una película del día actual	SM	Móvil	4	4	Hecho
4	Subtarea	Archivo PHP para devolver los horarios de una película del día siguiente	SM	Móvil	4	4	Hecho
5	Subtarea	Archivo PHP para devolver los horarios de una película del dos días después	SM	Móvil	3	3	Hecho
7	Subtarea	Archivo PHP para devolver los horarios de una película de las próximas dos horas	SM	Móvil	3	4	Hecho
8	Subtarea	Archivo PHP para devolver las películas listadas como próximos estrenos	SM	Móvil	4	4	Hecho
3, 4, 5 6, 7, 8	Subtarea	Función utilitaria para devolver los horarios de una película y cine en cualquier periodo de tiempo.	JG	Web	4	5	Hecho
	Creación de archivos y funciones de serialización de datos				12	13	Hecho
3, 4, 5 6, 7, 8	Subtarea	Archivo .JS para serializar datos de horarios de películas de todos los periodos de tiempo	SM	Móvil	4	4	Hecho
	Subtarea	Función para serializar y listar películas en cualquier periodo de tiempo	JG	Web	4	4	Hecho
	Subtarea	Función para serializar datos de horarios y cines de una película	JG	Web	4	5	Hecho

	Creación de plantillas de lado del cliente en todas las plataformas				12	14	Hecho
3, 4, 5 6, 7, 8	Subtare a	Plantilla HMTL para listar las películas que tengan horarios de cualquier periodo de tiempo	JG	Web	3	4	Hecho
	Subtare a	Plantilla HMTL para listar los horarios y cines de una película	JG	Web	3	1	Hecho
	Subtare a	Construcción de código HTML a partir de datos serializados en archivo .JS de películas	SM	Móvil	3	5	Hecho
	Subtare a	Archivo .JS para serializar y listar películas en carruseles	SM	Móvil	3	4	Hecho
3, 4, 5 6, 7, 8	Ingreso de datos de prueba en BDD		JG	BDD	2	2	Hecho
3, 4, 5 6, 7, 8	Pruebas de llamado de datos desde plantillas a archivos query en plataforma web		JG/SM	Web y Móvil	5	4	Hecho
3, 4, 5 6, 7, 8	Pruebas de presentación de datos en plantillas HTML		JG	Web	1	1	Hecho
3, 4, 5 6, 7, 8	Pruebas de presentación de datos en vistas		SM	Móvil	2	2	Hecho
3, 4, 5 6, 7, 8	Verificación de datos entre plantillas y BDD		JG/SM	Web y Móvil	2	2	Hecho
	Tareas no planificadas		JG/SM	Web/Móvil	5	5	Hecho

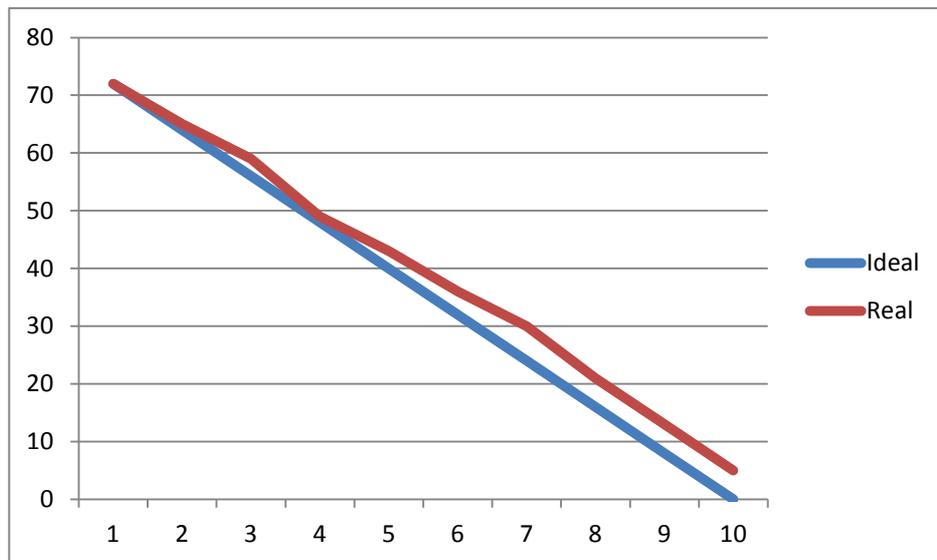


Figura 11. Gráfico Burndown sprint 1.

2.3.9.4. Demos

Sitio Web:

Intriga
Prisoners

GALIFICACIÓN DE LOS USUARIOS: ★★★★★

Duración: 153 minutos
Censura: 12 años
Género: Suspense
País de Origen: Estados Unidos

Director: Denis Villeneuve
Guionista: [no legible]

Elenco: Hugh Jackman, Jake Gyllenhaal, Viola Davis
La hija de seis años de Keller Dover, Anna, está perdida, y conforme pasan las horas, todos empiezan a entrar en pánico. La mejor pista que tienen es una casa rodante que había esta... [ver más](#)

Funciones para Hoy Cambiar Día ▾

Multicines CCI
Av. Amazonas y NNUJ
1800-ELCINE

SUBTITULADA 13:00 16:05 21:35

Multicines Condado
Av. Mariscal Sucre y J. Kennedy
1800-ELCINE

SUBTITULADA 15:20 20:45 21:40

Figura 12. Consulta de carteleras de una película en el sitio web.

Aplicación:



Figura 13. Consulta de cartelera de una película en la aplicación iOS.

2.3.9.5. Cierre y Retrospectiva

Al finalizar el primer sprint, como se puede ver en la figura 12 y 13 se ha implementado la consulta de cartelera de una película del presente día, del día siguiente, de pasado mañana, de la semana actual y de las próximas dos horas.

El sprint tuvo tareas que tomaron más tiempo del planificado y gracias al factor de dedicación propuesto se pudo ajustar agregando al sprint 5 horas de tareas no planificadas. En los siguientes sprints se deberá mejorar la planificación y el trabajo en equipo para optimizar el tiempo.

2.3.10. Segundo Sprint

2.3.10.1. Objetivo

Implementar la consulta de información general de una película y un complejo determinado.

2.3.10.2. Alcance

Este sprint tendrá una duración de dos semanas al final de las cuales se podrá consultar, mediante un demo funcional, toda la información relacionada con una película y un complejo determinados.

2.3.10.3. Lista de Sprint

Tabla 10. Lista del segundo sprint

User story	Tarea		Responsable	Plataforma	Estimado (h)	Real (h)	Estado
12 y 13	Creación de archivos y funciones query de llamado y consulta a BDD				8	9	Hecho
12	Subtarea	Incluir en archivos PHP de películas por periodo de tiempo, el llamado para la información general de una película	SM	Móvil	2	2	Hecho
13	Subtarea	Archivo PHP para devolver la información general de un complejo de cine por ciudad	SM	Móvil	2	2	Hecho
12	Subtarea	Incluir en función utilitaria de llamado de horarios de películas, el llamado para la información general de una película	JG	Web	2	3	Hecho
13	Subtarea	Incluir en función utilitaria de llamado de horarios de cines el llamado para la información general de una película	JG	Web	2	2	Hecho
12 y 13	Creación de archivos y funciones de serialización de datos				12	11	Hecho
12	Subtarea	Incluir en archivo vista .JS de películas la serialización de la información general de una película	SM	Móvil	3	3	Hecho
13	Subtarea	Incluir en archivo vista .JS de películas la serialización de la información general de un complejo de cines	SM	Móvil	3	4	Hecho
12	Subtarea	Incluir en la función vista de películas la serialización de la información general de una película	JG	Web	3	2	Hecho
13	Subtarea	Incluir en la función vista de películas la serialización de la información general de un complejo de cines	JG	Web	3	2	Hecho
12 y 13	Creación de plantillas de lado del cliente en todas las plataformas				8	11	Hecho
12	Subtarea	Incluir en archivo de presentación, bloque de información general de película	SM	Móvil	2	3	Hecho
13	Subtarea	Incluir en archivo de presentación, bloque de información general de complejo de cines	SM	Móvil	2	2	Hecho

12	Subtarea	Incluir en archivo HTML bloque de información general de película	JG	Web	2	2	Hecho
13	Subtarea	Incluir en archivo HTML bloque de información general de complejo de cines	JG	Web	2	4	Hecho
12, 13		Ingreso de datos de prueba en BDD	JG	BDD	2	1	Hecho
12, 13		Pruebas de llamado de datos desde plantillas a archivos query en plataforma web	JG/SM	Web y Móvil	5	7	Hecho
12, 13		Pruebas de presentación de datos en plantillas HTML	JG	Web	1	2	Hecho
12, 13		Pruebas de presentación de datos en vistas	SM	Móvil	2	2	Hecho
12, 13		Verificación de datos entre plantillas y BDD	JG/SM	Web y Móvil	2	1	Hecho
		Tareas no planificadas	JG/SM	Web/Móvil	4	4	Hecho

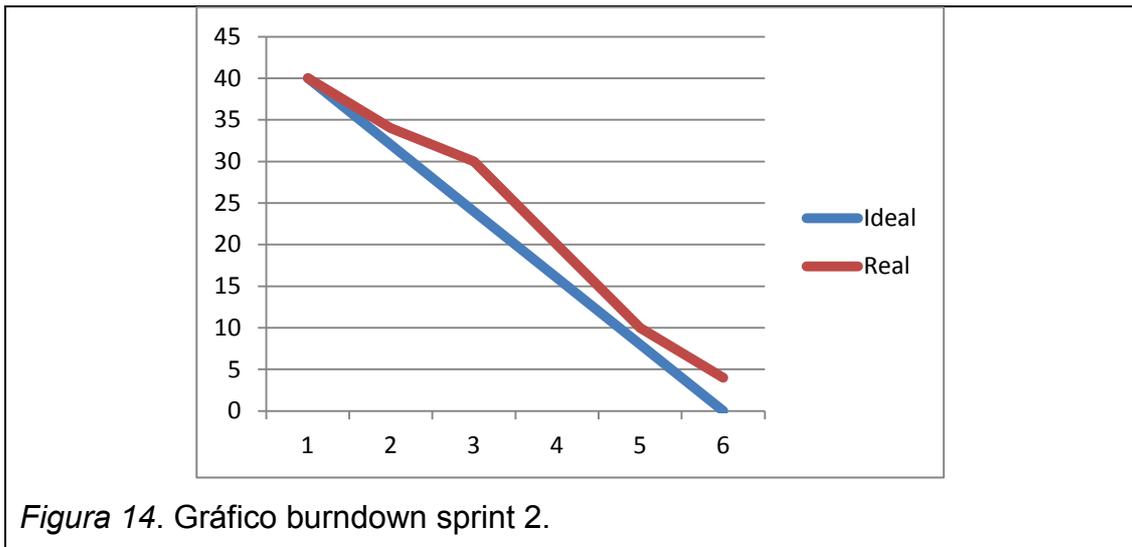


Figura 14. Gráfico burndown sprint 2.

2.3.10.4. Demos

Sitio Web

De Roma con amor
To Rome with Love

CALIFICACIÓN DE LOS USUARIOS
★★★★★

Duración: 112 minutos
Censura: 12 años
Género: Comedia
País de Origen: Estados Unidos

Director: Woody Allen
Elenco: Woody Allen, Penélope Cruz, Jesse Eisenberg

Cuatro historias independientes con un escenario común: la ciudad de Roma. En la primera, un matrimonio americano viaja a Italia para conocer a la familia del prometido de su hija. En la segunda, un italiano se hace famoso sin motivo de la noche a la mañana. En la tercera, un arquitecto californiano visita Roma con sus amigos donde conoce a un estudiante y, en la cuarta, una recién casada se pierde en la capital italiana, a la que ha ido a visitar a los familiares de su marido. [ocultar](#)

Figura 15. Consulta de la información general de una película en sitio web.

Cineplex
Tumbaco km. 14 1/2 en Ventura Mall
02 2372970

Funciones >

Figura 16. Consulta de la información general de un cine.

Aplicación:

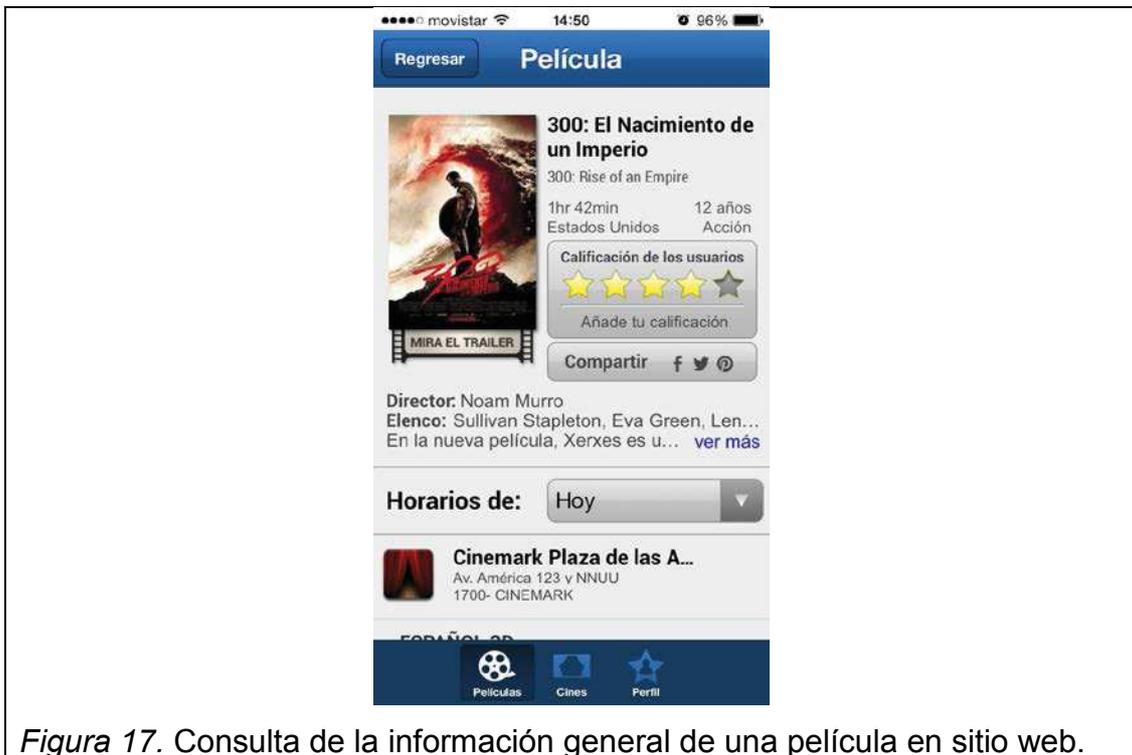


Figura 17. Consulta de la información general de una película en sitio web.



Figura 18. Consulta de la información general de un cine.

2.3.10.5. Cierre y Retrospectiva

Al finalizar el presente sprint se puede consultar la información de una película y de un cine como se observa en las figuras 15,16,17 y 18. Se realizó un ajuste en la planificación del sprint y a pesar del ajuste en la planificación se requirieron 4 horas de tareas no planificadas. Se recomienda planificar el próximo sprint tomando en cuenta tiempo adicional para tareas que puedan presentar problemas inesperados.

2.3.11. Tercer Sprint

2.3.11.1. Objetivo

Implementar la consulta de carteleras por complejos de cine en una ciudad determinada del presente día, del día siguiente y de pasado mañana.

2.3.11.2. Alcance

Este sprint tendrá una duración de dos semanas al final de las cuales se podrá consultar, mediante un demo funcional, las carteleras de todas las salas de cine en una ciudad predeterminada en los intervalos de tiempo determinados.

2.3.11.3. Lista de Sprint

Tabla 11. Lista del tercer sprint.

User story	Tarea	Responsable	Plataforma	Estimado (h)	Real (h)	Estado	
9, 10, 11	Creación de archivos y funciones query de llamado y consulta a BDD			12	13	Hecho	
9	Subtarea	Archivo PHP para devolver las películas y sus horarios de un cine del día actual	SM	Móvil	3	4	Hecho
10	Subtarea	Archivo PHP para devolver las películas y sus horarios de un cine del día siguiente	SM	Móvil	3	4	Hecho
11	Subtarea	Archivo PHP para devolver las películas y sus horarios de un cine de dos días después	SM	Móvil	3	2	Hecho
9, 10, 11	Subtarea	Función utilitaria para devolver los horarios de películas de un cine en cualquier periodo de tiempo	JG	Web	3	3	Hecho
9, 10, 11	Creación de archivos y funciones de serialización de datos			12	12	Hecho	
9, 10, 11	Subtarea	Archivo .JS para serializar datos de horarios de películas de un cine de todos los periodos de tiempo	SM	Móvil	5	6	Hecho
	Subtarea	Función para serializar y listar los complejos de cine	JG	Web	3	2	Hecho
	Subtarea	Función para serializar datos de horarios y películas de un cine	JG	Web	4	4	Hecho
9, 10, 11	Creación de plantillas de lado del cliente en todas las plataformas			12	15	Hecho	
9, 10, 11	Subtarea	Plantilla HTML para listar todos los complejos de cine de una ciudad	JG	Web	3	4	Hecho
	Subtarea	Plantilla HTML para listar los horarios y películas de un complejo de cines	JG	Web	3	4	Hecho
	Subtarea	Construcción de código HTML a partir de datos serializados en archivo .JS de complejos	SM	Móvil	3	4	Hecho
	Subtarea	Archivo .JS para serializar y listar todos los complejos de cine de una ciudad	SM	Móvil	3	3	Hecho
9, 10, 11	Ingreso de datos de prueba en BDD		JG	BDD	2	2	Hecho
9, 10, 11	Pruebas de llamado de datos desde plantillas a archivos query		JG/SM	Web y Móvil	5	4	Hecho
9, 10, 11	Pruebas de presentación de datos en plantillas HTML		JG	Web	1	1	Hecho
9, 10, 11	Pruebas de presentación de datos en vistas		SM	Móvil	2	2	Hecho
9, 10, 11	Verificación de datos entre plantillas y BDD		JG/SM	Web y Móvil	2	2	Hecho
	Tareas no planificadas		JG/SM	Web/Móvil	3	3	Hecho

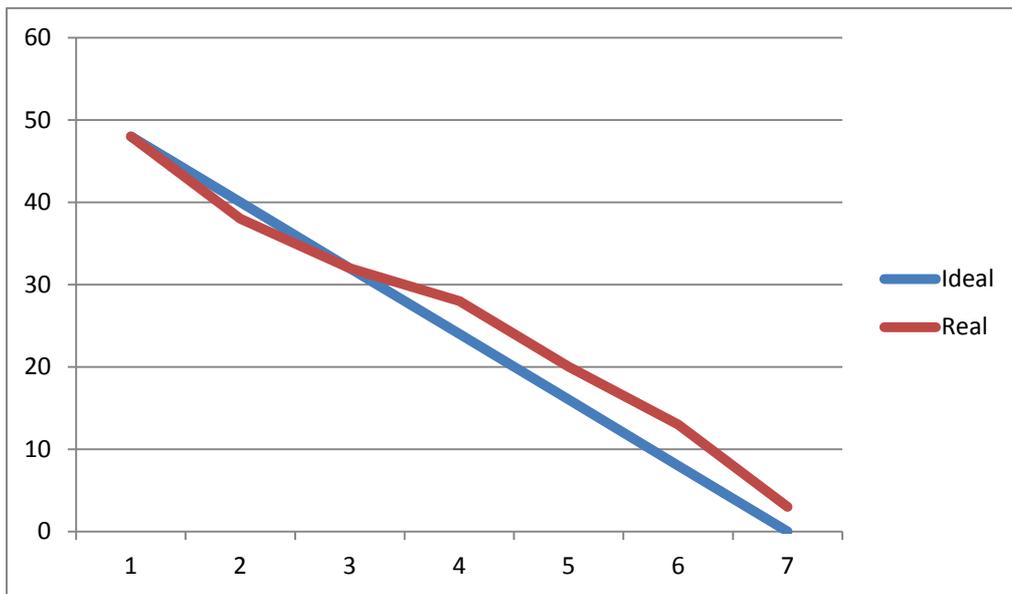


Figura 19. Gráfico burndown sprint 3.

2.3.11.4. Demos

Sitio Web:

Multicines Condado
Av. Mariscal Sucre y J. Kennedy
1800-ELCINE

Añadir a tus complejos favoritos

Funciones para Hoy Cambiar Día ▾

Escándalo Americano
American Hustle

Duración: 138 minutos
Censura: 16
Género: Drama

Más información...

SUBTITULADA

13:05 16:15 21:30 13:10
18:40 21:30

Figura 20. Consulta de horarios por complejos de cine en sitio web.

Aplicación:

The screenshot shows a mobile application interface for finding movie theaters and showtimes. At the top, the status bar displays 'movistar', signal strength, Wi-Fi, time '14:56', and battery '96%'. Below the status bar is a blue header with a 'Regresar' button and the title 'Complejos'. The main content area features a theater listing for 'Supercines San Luis' with a red curtain icon, address 'Av. San Luis S/N e Isla Santa Clara, Edif. San Luis Shopping Local 213 Nivel 2', and phone number '593-2-2090-015'. A red button with a heart icon says 'Añadir a tus complejos favoritos'. Below this is a 'Horarios de:' section with a dropdown menu set to 'Hoy'. The first movie listing is 'Quito 2023' with a poster showing a raised fist, a duration of '1hr 29min', and the language 'ESPAÑOL' with a showtime of '17:40'. The second listing is 'Saudade' with a poster of a woman on a bicycle. At the bottom is a dark blue navigation bar with icons and labels for 'Películas', 'Cines', and 'Perfil'.

●●●●○ movistar 14:56 96%

Regresar Complejos

 **Supercines San Luis**
Av. San Luis S/N e Isla Santa Clara, Edif.
San Luis Shopping Local 213 Nivel 2
593-2-2090-015

Añadir a tus complejos favoritos 

Horarios de: Hoy 

 **Quito 2023**
Quito 2023
1hr 29min

ESPAÑOL
17:40

 **Saudade**
Saudade

 Películas  Cines  Perfil

Figura 21. Consulta de horarios por complejos de cine en sitio web.

2.3.11.5. Cierre y Retrospectiva

Gracias a que el primer sprint es estructuralmente muy parecido al tercero en el aspecto técnico, no se presentaron mayores complicaciones al término del mismo, como se puede observar en las figuras 20 y 21 se puede consultar los horarios de un cine en una ciudad predeterminada, para el día de hoy mañana o pasado. Este sprint pudo ser completado agregando 3 horas de tareas no planificadas lo que indica que la planificación aún puede mejorar para próximos sprints.

2.3.12. Cuarto Sprint

2.3.12.1. Objetivo

Implementar el registro de usuarios sociales e ingreso de usuarios anónimos en el sistema.

2.3.12.2. Alcance

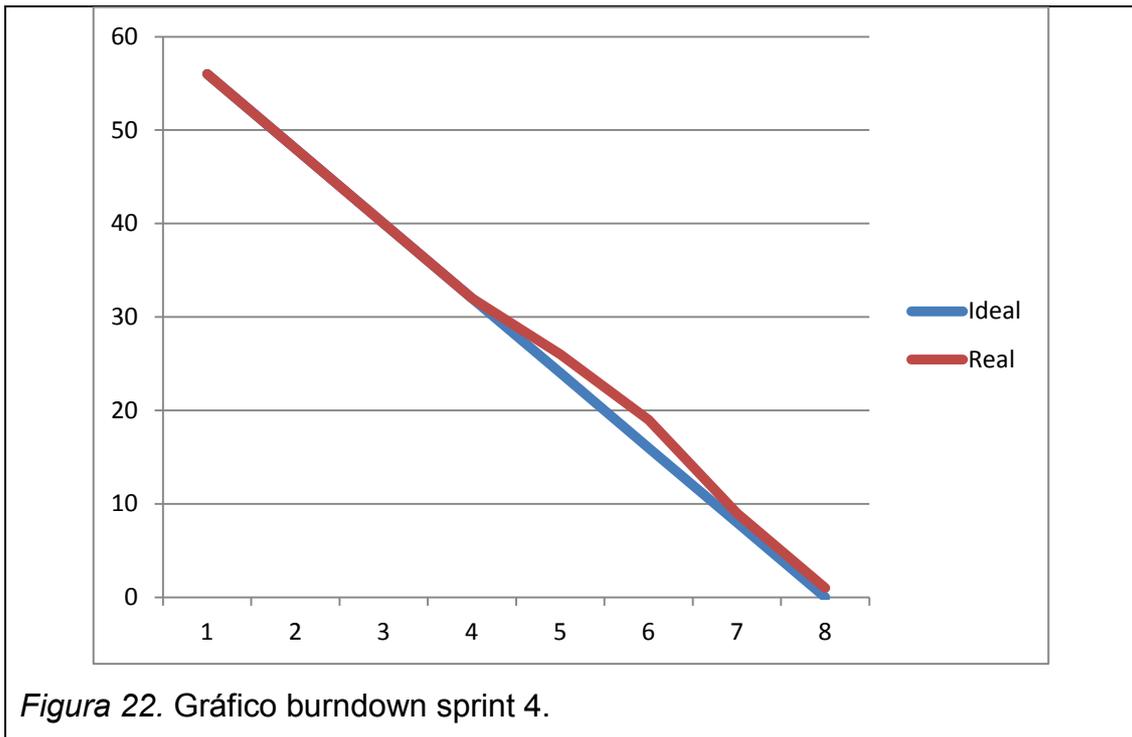
Este sprint tendrá una duración de dos semanas al final de las cuales se podrá registrar usuarios nuevos mediante el API de registro de Facebook e ingresar anónimamente al sistema.

En este sprint se implementará también las restricciones de funcionalidades para usuarios anónimos.

2.3.12.3. Lista de Sprint

Tabla 12. Lista del cuarto sprint.

User story	Tarea	Responsable	Plataforma	Estimado (h)	Real (h)	Estado
1	Creación de cuenta y aplicación en plataforma de desarrolladores de Facebook	JG	Web/Móvil	4	1	Hecho
1	Creación de funciones de inicialización JavaScript con SDK y AppID de Facebook			12	10	Hecho
1	Subtarea Función de llamado asincrónico a API de Facebook	JG	Web	3	3	Hecho
1	Subtarea Función de registro con API de Facebook	JG	Web	3	3	Hecho
1	Subtarea Función de llamado asincrónico a API de Facebook	SM	Móvil	3	2	Hecho
1	Subtarea Función de registro con API de Facebook	SM	Móvil	3	2	Hecho
1	Creación de funciones y queries de inserción de usuarios a BDD			8	10	Hecho
1	Subtarea Función de ingreso de usuario mediante POST Ajax hacia API propio	JG	Web	4	5	Hecho
1	Subtarea Archivo PHP de inserción de usuario y habilitación de API para plataforma Web	SM	Móvil/API	4	5	Hecho
1	Creación de funciones de respuesta, manejo de errores y redirección depuse de registro			8	10	Hecho
1	Subtarea Función que recarga la página y coloca nombre de usuario en plantilla base	JG	Web	4	5	Hecho
1	Subtarea Función que redirecciona a la App de Facebook para aceptación y regresa con datos de usuario	SM	Móvil	4	5	Hecho
1	Creación de funciones de comprobación de registro para restricción a anónimos			8	9	Hecho
1	Subtarea Función que comprueba el estado del usuario para permitir acceso a funcionalidad o mostrar mensaje de registro con Facebook	JG	Web	4	5	Hecho
1	Subtarea Función que comprueba el estado del usuario para permitir acceso a funcionalidad o mostrar mensaje de registro con Facebook	SM	Móvil	4	4	Hecho
1	Creación de botones y función JavaScript de comprobación de status y permisos	SM	Móvil	3	3	Hecho
1	Creación de botones y función JavaScript de comprobación de status y permisos	JG	Web	3	3	Hecho
1	Pruebas de registro con Facebook y comprobación en BDD	JG	Web	3	3	Hecho
1	Pruebas de registro con Facebook y comprobación en BDD	SM	Móvil	3	4	Hecho
2	Creación de funciones de redirección para usuarios anónimos	SM	Móvil	3	4	Hecho
	Tareas no planificadas	SM	Móvil	1	1	Hecho



2.3.12.4. Demos

Sitio Web:



Aplicación:

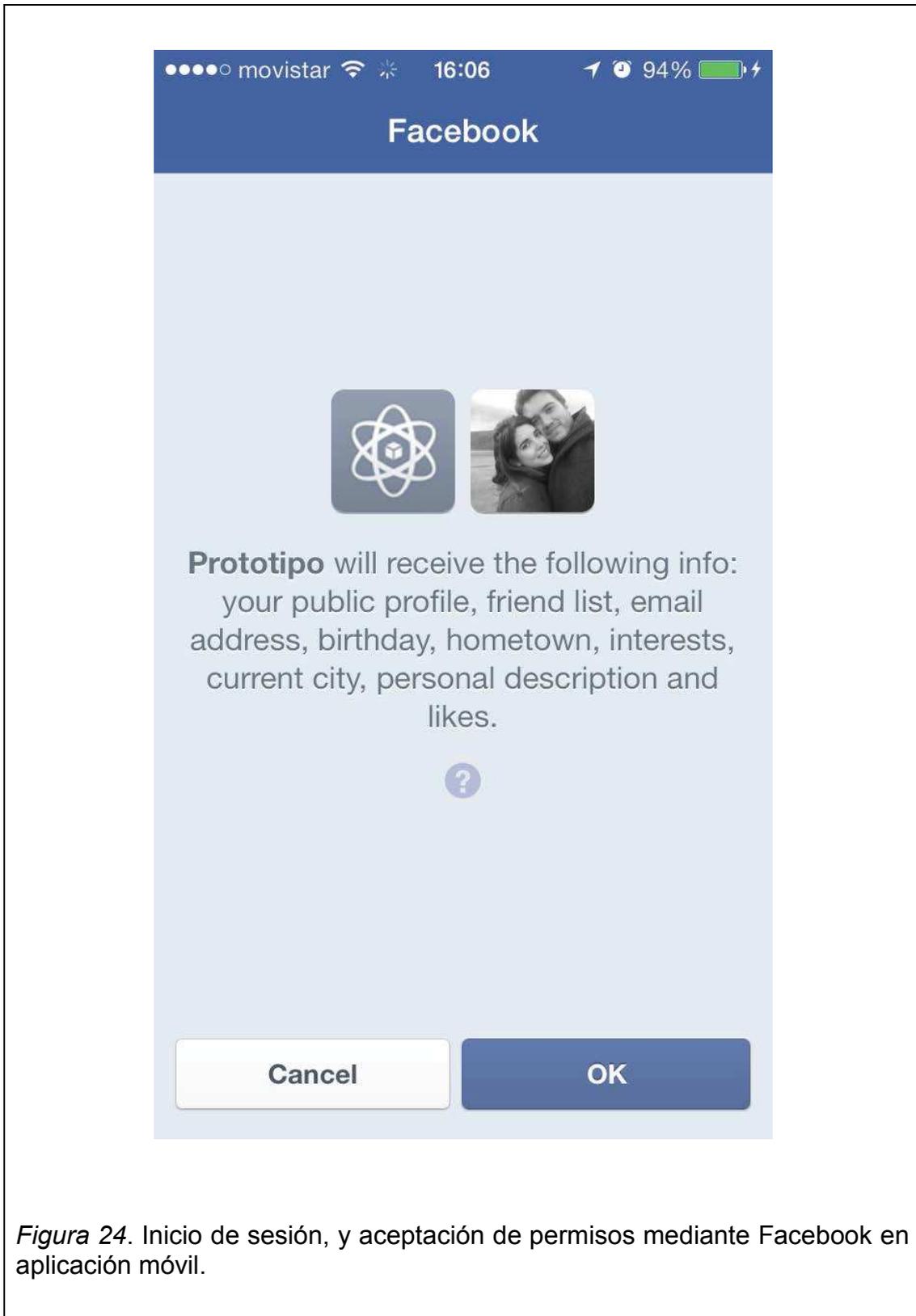


Figura 24. Inicio de sesión, y aceptación de permisos mediante Facebook en aplicación móvil.

2.3.12.5. Cierre y Retrospectiva

Se implementó el registro de usuarios sociales e ingreso de usuarios anónimos en el sistema como se puede ver en las figuras 23 y 24.

Este sprint se desarrolló, con excepción de ciertos contratiempos, de manera correcta. Se completó en un tiempo óptimo teniendo que aumentar una hora de tarea no planificada.

2.3.13. Quinto Sprint

2.3.13.1. Objetivo

Implementar la funcionalidad de calificación de una película para usuarios sociales y la consulta de la misma en la sección de perfil.

2.3.13.2. Alcance

Este sprint tendrá una duración de dos semanas al final de las cuales, mediante un demo funcional, se podrá calificar (1 a 5 estrellas) una película que se encuentre en cartelera y consultar las calificaciones realizadas en la sección de perfil

2.3.13.3. Lista de Sprint

Tabla 13. Lista del quinto sprint.

User story	Tarea	Responsable	Plataforma	Estimado (h)	Real (h)	Estado		
14	Generación de código HTML desde archivos de vistas	SM	Móvil	2	2	Hecho		
14	Inclusión de código HTML en plantillas	JG	Web	3	3	Hecho		
14	Creación de gráficos de estrellas	JG	Web/Móvil	4	4	Hecho		
14	Creación de botones para calificaciones	JG	Web/Móvil	4	4	Hecho		
14	Creación de función JavaScript para publicar la calificación mediante el API de Facebook	JG	Web	4	4	Hecho		
14	Creación de disparador de función en HTML y enlace a función de comprobación de registro	JG	Web	4	4	Hecho		
14	Creación de función JavaScript para publicar la calificación mediante el API de Facebook	SM	Móvil	4	4	Hecho		
14	Creación de disparador de función en HTML y enlace a función de comprobación de registro	SM	Móvil	4	4	Hecho		
14	Creación de archivos y funciones query para el ingreso de datos a BDD			8	8	Hecho		
14	Subtarea	Función de ingreso de calificación mediante POST Ajax hacia API propio		JG	Web	4	4	Hecho
14	Subtarea	Archivo PHP de inserción de calificación y habilitación de API para plataforma Web		SM	Móvil/API	4	4	Hecho
17	Creación de archivos y funciones query de llamado y consulta a BDD			8	8	Hecho		
17	Subtarea	Archivo PHP para devolver las calificaciones (película y estrellas) de un usuario		SM	Móvil	4	4	Hecho
17	Subtarea	Función utilitaria para devolver todos los datos de un usuario (modulo calificaciones)		JG	Web	4	4	Hecho
17	Creación de archivos y funciones de serialización de datos			6	6	Hecho		
17	Subtarea	Archivo .JS para serializar datos de la sección de perfil de usuario (modulo calificaciones)		SM	Móvil	3	3	Hecho
17	Subtarea	Función para serializar datos de la sección de perfil de usuario (modulo calificaciones)		JG	Web	3	3	Hecho
14, 17	Pruebas de llamado de datos desde plantillas a archivos query		JG/SM	Web y Móvil	4	4	Hecho	
14	Prueba de calificación y verificación de datos en BDD y en Facebook		JG	Web y Móvil	3	3	Hecho	
17	Pruebas de presentación de datos en plantillas HTML		JG	Web	2	2	Hecho	
17	Pruebas de presentación de datos en vistas		SM	Móvil	2	2	Hecho	
17	Verificación de datos entre plantillas y BDD		JG/SM	Web y Móvil	2	2	Hecho	

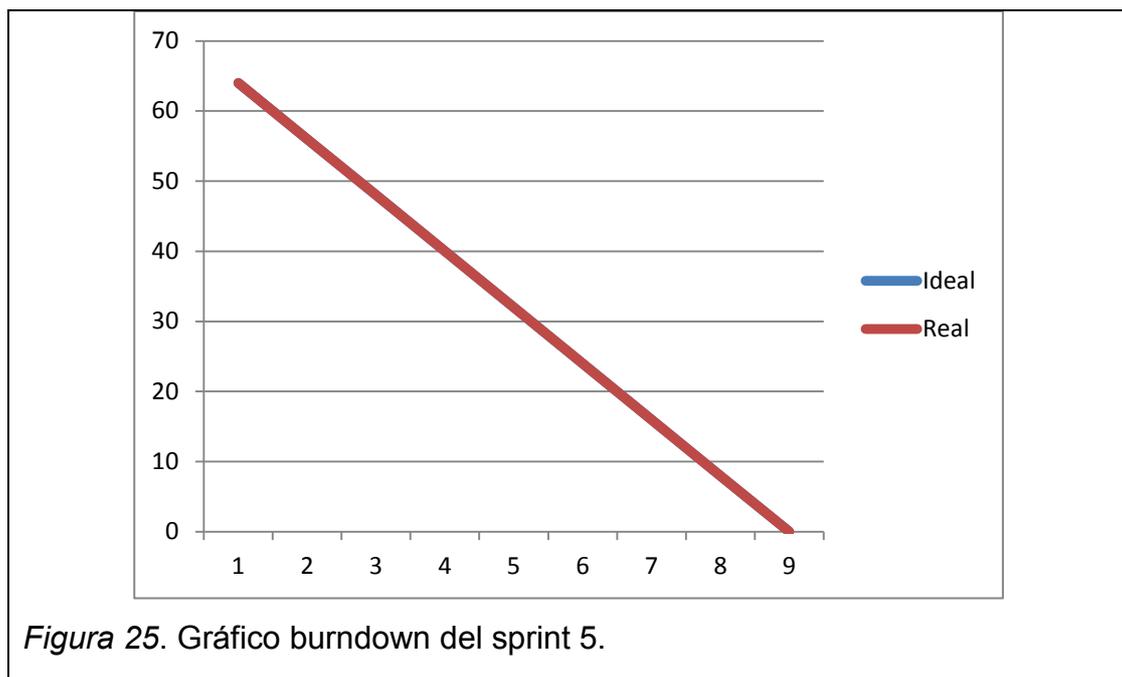


Figura 25. Gráfico burndown del sprint 5.

2.3.13.4. Demos

Sitio Web:



Figura 26. Capturas de pantalla del proceso de calificación de una película.



Figura 27. Consulta de una calificación en la sección de perfil.

Aplicación:

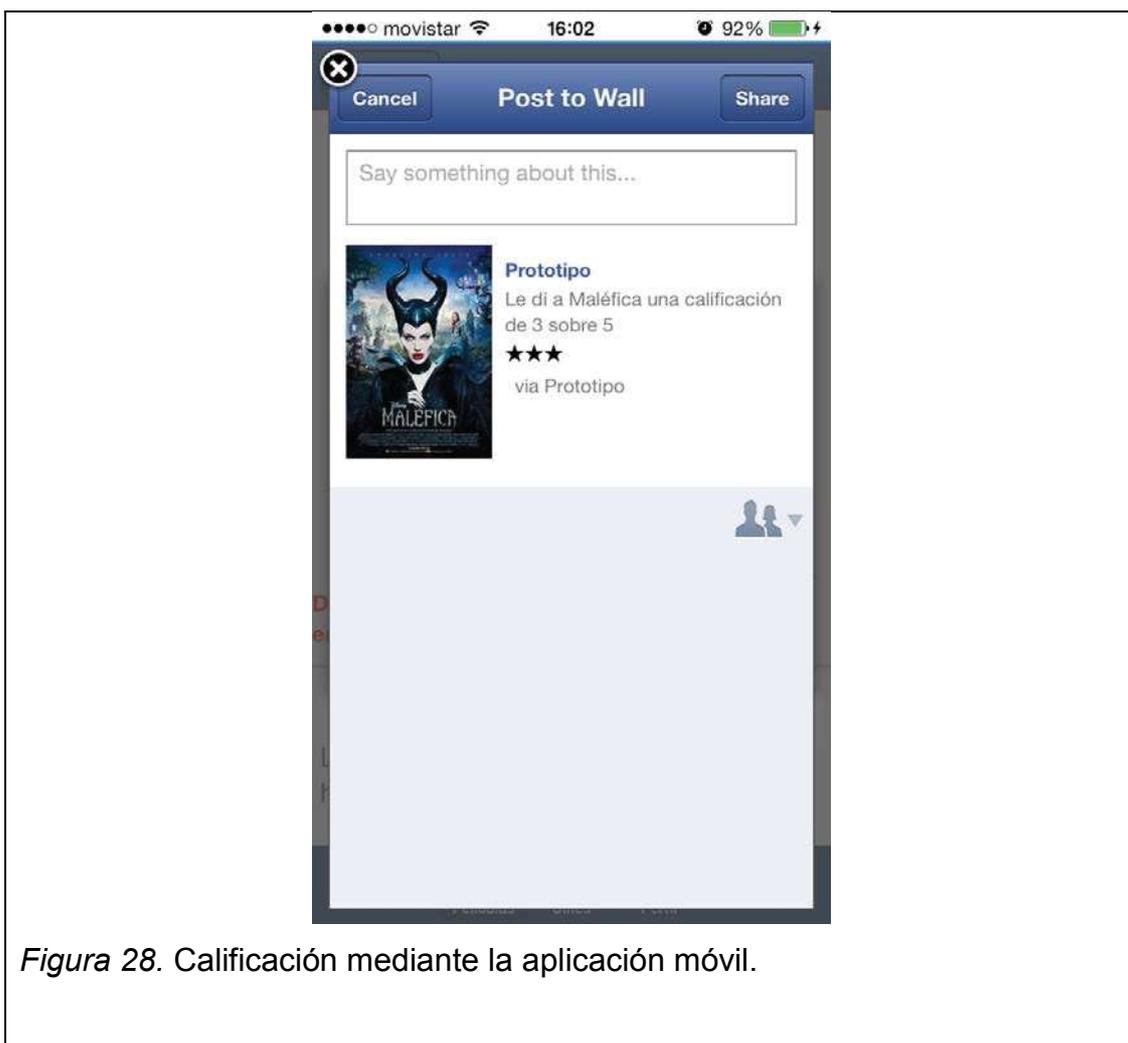


Figura 28. Calificación mediante la aplicación móvil.

2.3.13.5. Cierre y Retrospectiva

Como se observa en las figuras 26,27 y 28 se logró implementar la funcionalidad de calificación de una película para usuarios sociales. Tomando en cuenta la experiencia adquirida en la planificación de sprints anteriores, este sprint se realizó de manera correcta en el tiempo esperado.

2.3.14. Sexto Sprint

2.3.14.1. Objetivo

Implementar la funcionalidad para guardar una película en estrenos esperados y la consulta de la misma en la sección de perfil.

2.3.14.2. Alcance

Este sprint tendrá una duración de dos semanas al final de las cuales, mediante un demo funcional, se podrá guardar una película que se encuentre en la sección de próximos estrenos a una lista de estrenos esperados y consultar posteriormente dicha lista en la sección de perfil.

2.3.14.3. Lista de Sprint

Tabla 14. Lista del sexto sprint.

User story	Tarea	Responsable	Plataforma	Estimado (h)	Real (h)	Estado	
15	Generación de código HTML desde archivos de vistas	SM	Móvil	3	3	Hecho	
15	Inclusión de código HTML en plantillas	JG	Web	3	3	Hecho	
15	Creación de botones para guardar película	JG	Web/Móvil	4	4	Hecho	
15	Creación de función JavaScript para publicar la película mediante el API de Facebook	JG	Web	4	4	Hecho	
15	Creación de disparador de función en HTML y enlace a función de comprobación de registro	JG	Web	4	4	Hecho	
15	Creación de función JavaScript para publicar la película mediante el API de Facebook	SM	Móvil	4	4	Hecho	
15	Creación de disparador de función en HTML y enlace a función de comprobación de registro	SM	Móvil	4	4	Hecho	
15	Creación de archivos y funciones query para el ingreso de datos a BDD			8	8	Hecho	
15	Subtarea	Función de ingreso de calificación mediante POST Ajax hacia API propio	JG	Web	4	4	Hecho
15	Subtarea	Archivo PHP de inserción de usuario y habilitación de API para plataforma Web	SM	Móvil/API	4	4	Hecho
18	Creación de archivos y funciones query de llamado y consulta a BDD			8	8	Hecho	
	Subtarea	Archivo PHP para devolver las películas guardadas como estrenos esperados de un usuario	SM	Móvil	4	4	Hecho
	Subtarea	Función utilitaria para devolver todos los datos de un usuario (modulo estrenos esperados)	JG	Web	4	4	Hecho
18	Creación de archivos y funciones de serialización de datos			8	8	Hecho	
	Subtarea	Archivo .JS para serializar datos de la sección de perfil de usuario (modulo estrenos esperados)	SM	Móvil	4	4	Hecho
	Subtarea	Función para serializar datos de la sección de perfil de usuario (modulo estrenos esperados)	JG	Web	4	4	Hecho
15, 18	Pruebas de llamado de datos desde plantillas a archivos query		JG/SM	Web/Móvil	4	4	Hecho
15	Prueba de publicación y verificación de datos en BDD y en Facebook		JG	Web/Móvil	3	3	Hecho
18	Pruebas de presentación de datos en plantillas HTML		JG	Web	3	3	Hecho
18	Pruebas de presentación de datos en vistas		SM	Móvil	2	2	Hecho
18	Verificación de datos entre plantillas y BDD		JG/SM	Web/Móvil	2	2	Hecho

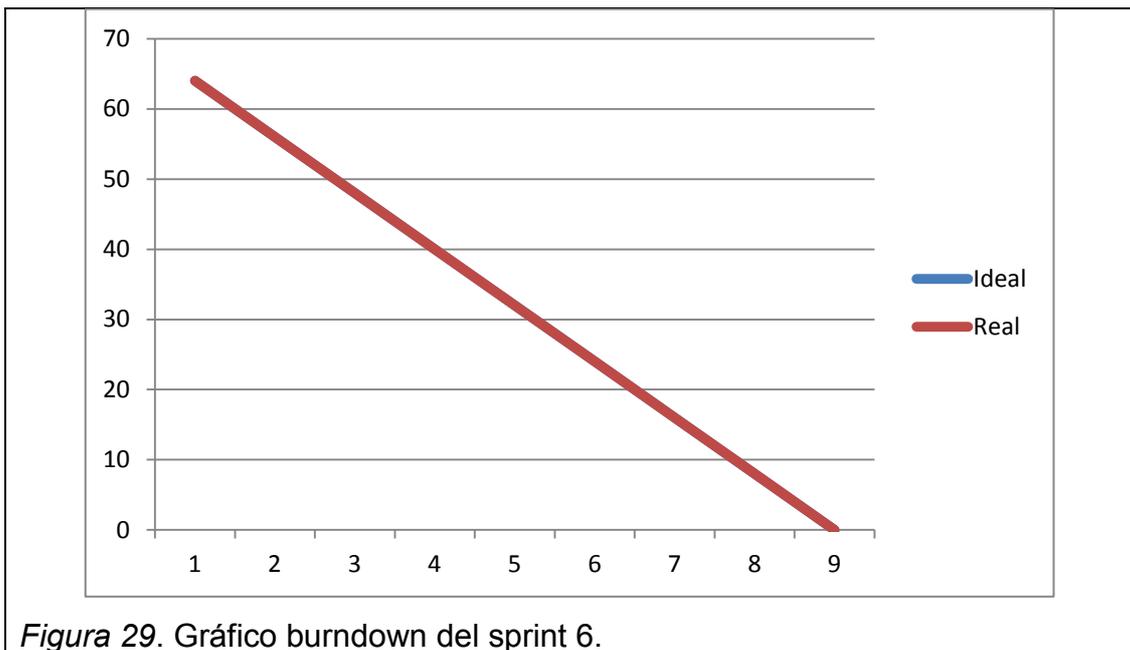


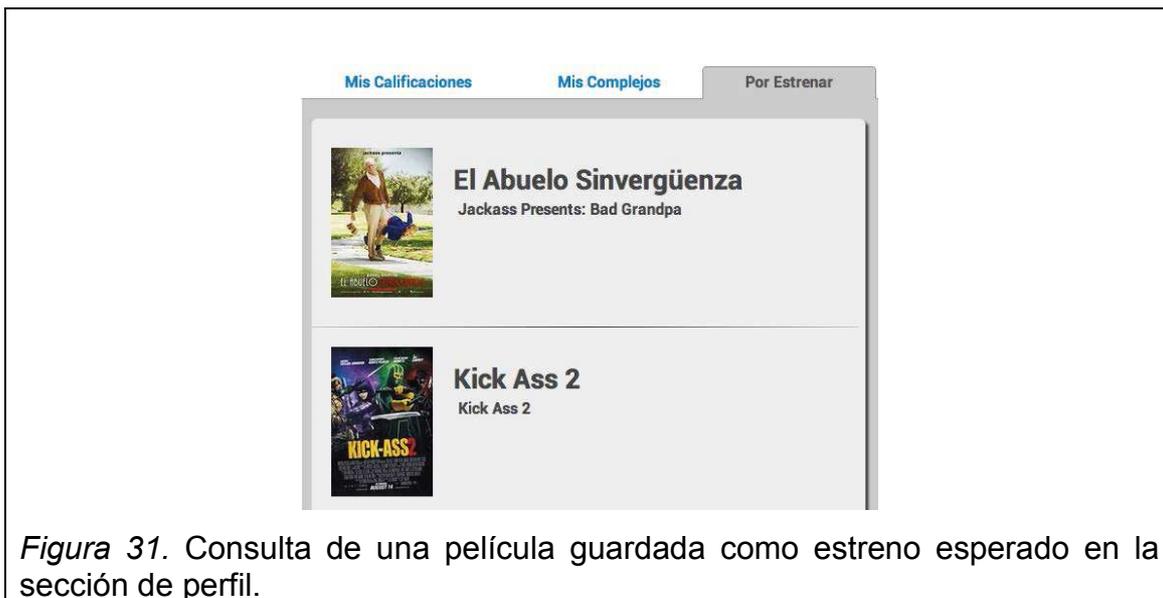
Figura 29. Gráfico burndown del sprint 6.

2.3.14.4. Demos

Sitio Web:



Figura 30. Proceso de guardado de un estreno esperado.



Aplicación:



2.3.14.5. Cierre y Retrospectiva

El sprint se llevó a cabo sin ningún contratiempo cumpliendo con lo planificado en el tiempo esperado.

Como se ve en las figuras 30, 31 y 32 se completó con éxito la funcionalidad para guardar y compartir un estreno esperado.

2.3.15. Séptimo Sprint

2.3.15.1. Objetivo

Implementar la funcionalidad para guardar un complejo de cine en la lista de favoritos y la consulta de la misma en la sección de perfil.

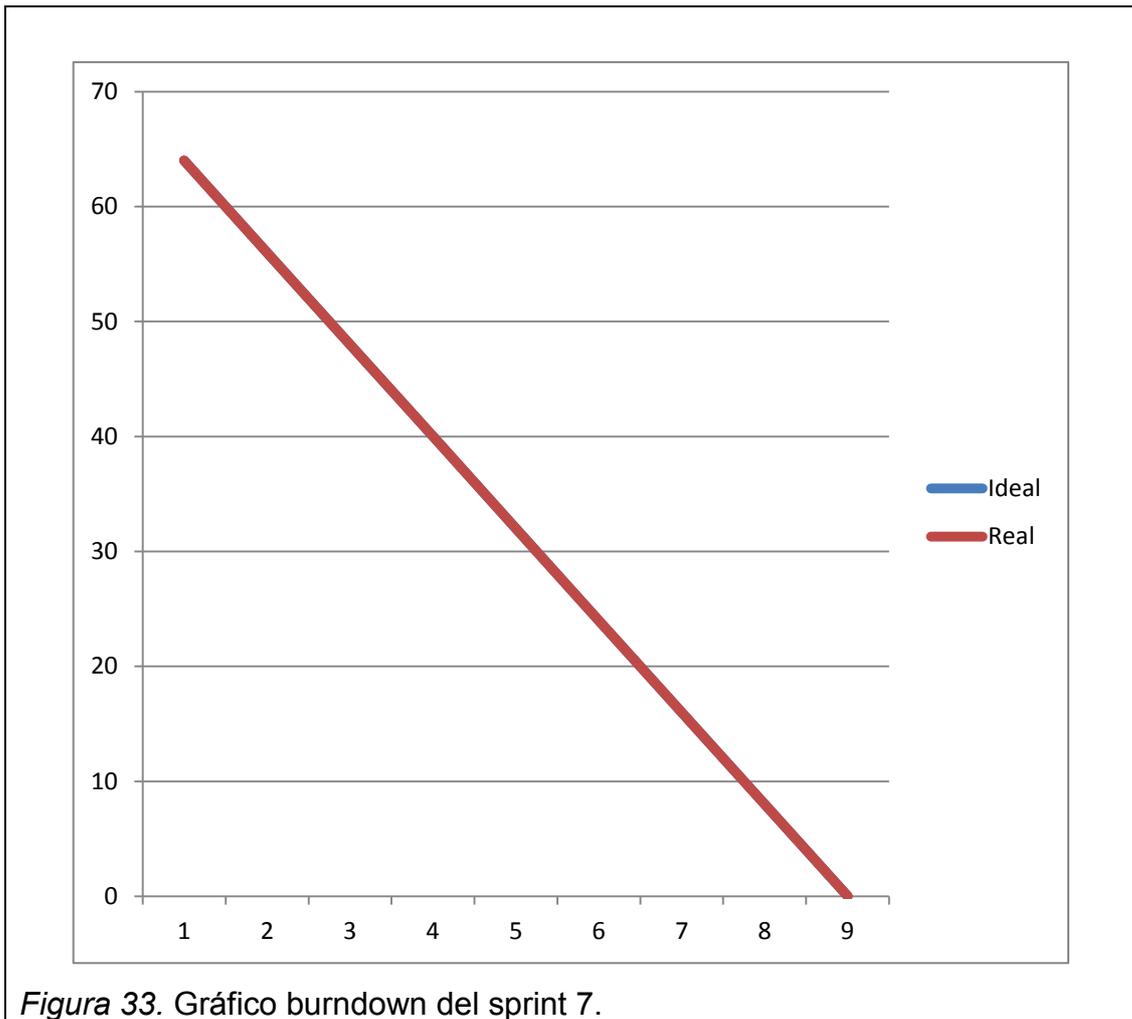
2.3.15.2. Alcance

Este sprint tendrá una duración de dos semanas al final de las cuales, mediante un demo funcional, se podrá guardar un cine a una lista de favoritos y consultar posteriormente dicha lista en la sección de perfil y en la lista general de cines.

2.3.15.3. Lista de Sprint

Tabla 15. Lista del séptimo sprint.

User story	Tarea	Responsable	Plataforma	Estimado (h)	Real (h)	Estado	
16	Generación de código HTML desde archivos de vistas	SM	Móvil	3	3	Hecho	
16	Inclusión de código HTML en plantillas	JG	Web	3	3	Hecho	
16	Creación de botones para guardar cine como favorito	JG	Web/Móvil	4	4	Hecho	
16	Creación de disparador de función en HTML y enlace a función de comprobación de registro	JG	Web	4	4	Hecho	
16	Creación de disparador de función en HTML y enlace a función de comprobación de registro	SM	Móvil	4	4	Hecho	
16	Creación de archivos y funciones query para el ingreso de datos a BDD			8	8	Hecho	
16	Subtarea	Función de ingreso de cine favorito mediante POST Ajax hacia API propio	JG	Web	4	4	Hecho
16	Subtarea	Archivo PHP de inserción de cine favorito y habilitación de API para plataforma Web	SM	Móvil/API	4	4	Hecho
19	Creación de archivos y funciones query de llamado y consulta a BDD			8	8	Hecho	
19	Subtarea	Archivo PHP para devolver los cines guardados como favoritos por un usuario	SM	Móvil	4	4	Hecho
19	Subtarea	Función utilitaria para devolver todos los datos de un usuario (modulo cines favoritos)	JG	Web	4	4	Hecho
19	Creación de archivos y funciones de serialización de datos			8	8	Hecho	
19	Subtarea	Archivo .JS para serializar datos de la sección de perfil de usuario (modulo cines favoritos)	SM	Móvil	4	4	Hecho
19	Subtarea	Función para serializar datos de la sección de perfil de usuario (modulo cines favoritos)	JG	Web	4	4	Hecho
19	Creación de archivos y funciones query para la eliminación de datos en BDD			6	6	Hecho	
19	Subtarea	Función de eliminación de cine favorito seleccionado	SM	Móvil	3	3	Hecho
19	Subtarea	Función de eliminación de cine favorito seleccionado	JG	Web	3	3	Hecho
19	Creación de función para desplegar cuadro de dialogo para eliminar cine favorito	SM	Móvil	3	3	Hecho	
19	Creación de función para desplegar cuadro de dialogo para eliminar cine favorito	JG	Web	3	3	Hecho	
16, 19	Pruebas de llamado de datos desde plantillas a archivos query	JG/SM	Web y Móvil	4	4	Hecho	
19	Pruebas de presentación de datos en plantillas HTML	JG	Web	2	2	Hecho	
19	Pruebas de presentación de datos en vistas	SM	Móvil	2	2	Hecho	
19	Verificación de datos entre plantillas y BDD	JG/SM	Web y Móvil	2	2	Hecho	



2.3.15.4. Demos

Sitio Web:



Figura 34. Sección para guardar un cine como favorito.



Figura 35. Consulta de un cine guardado como favorito través de la sección de perfil.



Figura 36. Consulta de un cine guardado como favorito través de la lista general de cines.

Aplicación:



Figura 37. Captura de pantalla de la sección para guardar un cine como favorito en la aplicación móvil.



Figura 38. Consulta de un cine guardado como favorito través de la lista general de cines en la aplicación móvil.

2.3.15.5. Cierre y Retrospectiva

El séptimo sprint se completó con éxito permitiendo agregar a un complejo como favorito esto se puede ver en las figuras 35, 36, 37 y 38.

En el aspecto técnico este sprint fue muy similar al anterior por lo que en la planificación del mismo se pudo estimar óptimamente el tiempo necesario para completar sus tareas.

2.3.16. Octavo Sprint

2.3.16.1. Objetivo

Implementar la funcionalidad para simular la compra de un boleto para una función de cine.

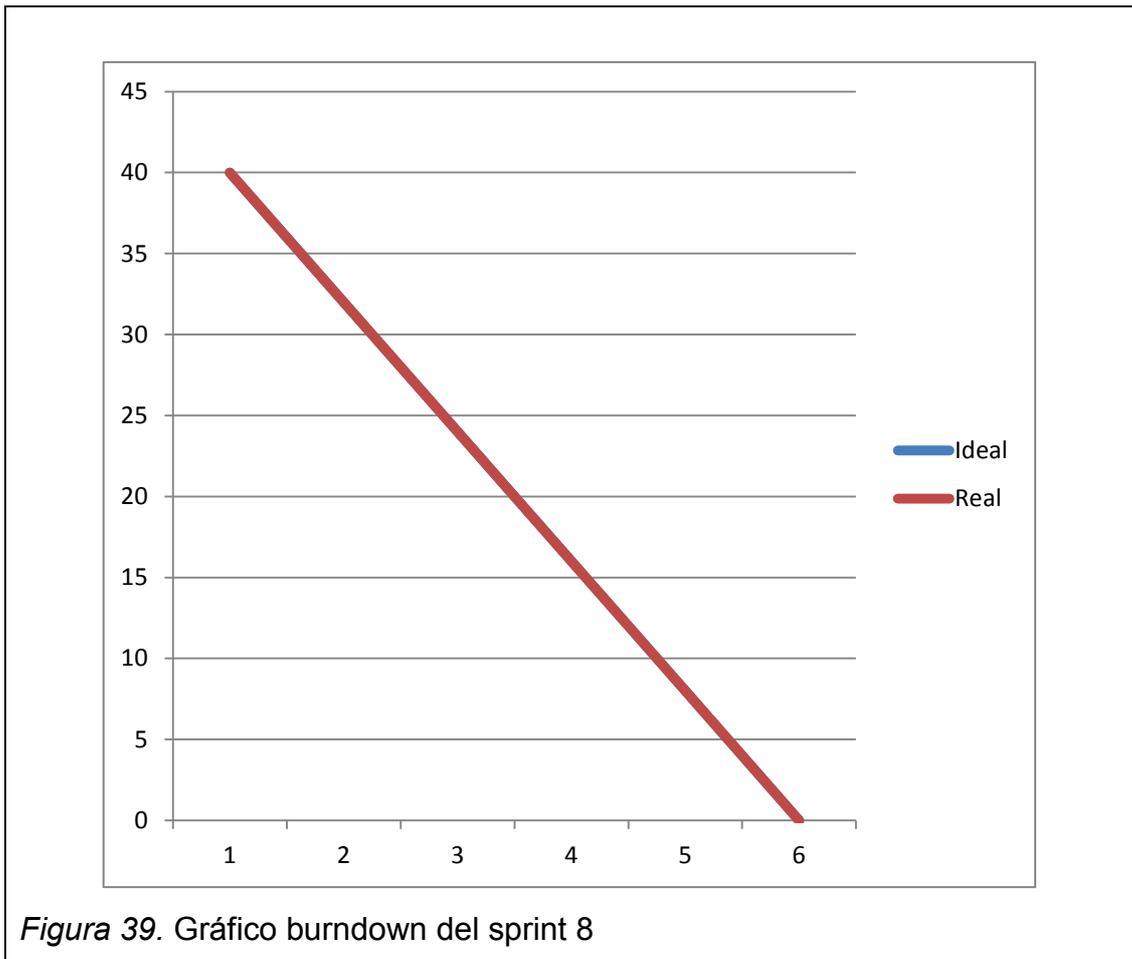
2.3.16.2. Alcance

Este sprint tendrá una duración de dos semanas al final de las cuales, mediante un demo funcional, se podrá simular la compra de un boleto para asistir a cualquier función que se encuentre en cartelera.

2.3.16.3. Lista de Sprint

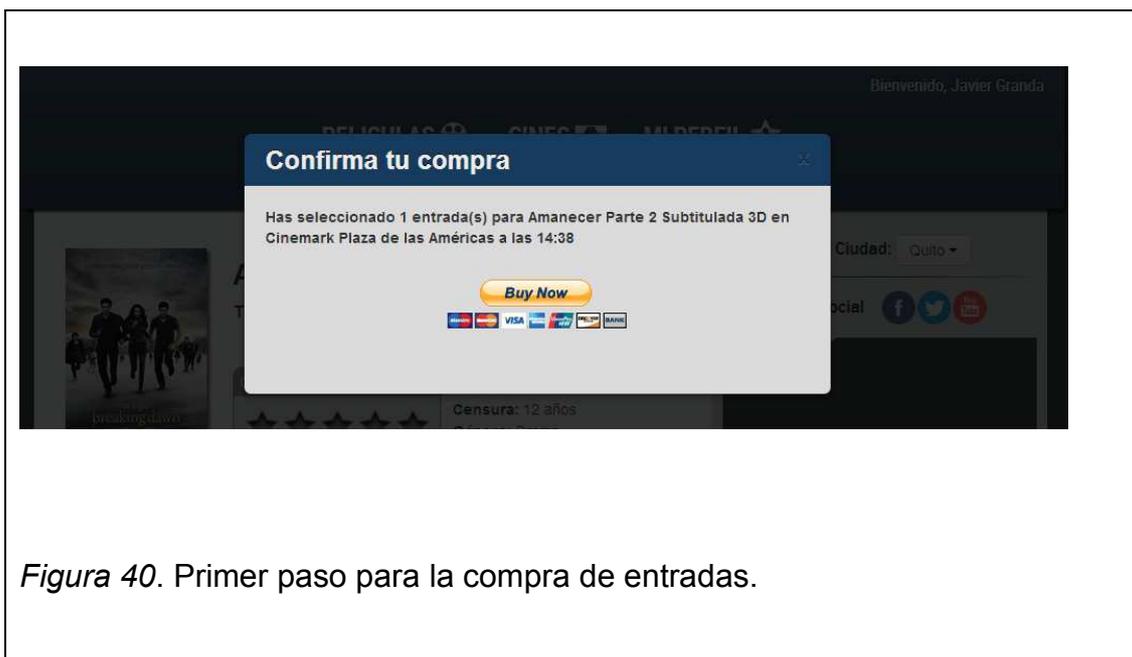
Tabla 16. Lista del octavo sprint.

User story	Tarea	Responsable	Plataforma	Estimado (h)	Real (h)	Estado
20	Crear cuenta de PayPal Sandbox	JG	Web/Móvil	2	2	Hecho
20	Creación de vistas de presentación de proceso de compra			8	8	Hecho
	Subtarea Crear vista para selección de opciones (idioma, tecnología, hora, cantidad) y su disparador	SM	Móvil	4	4	Hecho
	Subtarea Crear modal para confirmación de datos de compra y su disparador	JG	Web	4	4	Hecho
20	Creación de archivos y funciones query para el ingreso de datos a BDD			8	8	Hecho
	Subtarea Creación de archivo PHP de ingreso de registro de compra y habilitación de API para plataforma Web	SM	Móvil/API	4	4	Hecho
	Subtarea Función de ingreso de registro de compra mediante POST Ajax hacia API propio	JG	Web	4	4	Hecho
	Comunicación con API PayPal			8	8	Hecho
	Subtarea Creación de función de manejo de API de PayPal para ingreso de método de pago y finalización de pago	SM	Móvil	4	4	Hecho
	Subtarea Creación de función de envío de parámetros a URL de PayPal para confirmación de pago	JG	Web	4	4	Hecho
	Creación de función para recibir callback y confirmar datos de compra o alertar de errores	SM	Móvil	4	4	Hecho
	Creación de función para recibir callback y confirmar datos de compra o alertar de errores	JG	Web	4	4	Hecho
	Verificación de vista de selección de opciones con datos correctos	SM	Móvil	3	3	Hecho
	Prueba de compra y verificación en BDD	JG	Web/Móvil	3	3	Hecho



2.3.16.4. Demos

Sitio Web:



Your order summary

Descriptions	Amount
Entrada Cinemark Plaza de las Américas...	\$6.00
Item price: \$6.00	
Quantity: 1	
Item total	\$6.00
Tax:	\$0.72
	Total \$6.72 USD

Review your information

Pay Now

Shipping address [Change](#)

Sebastian Mansfield
1 Main St
San Jose, CA 95131
United States

Use as preferred shipping address

Note to seller: [Add](#)

Payment methods [Change](#) [Now accepting prepaid gift cards](#)

PayPal Balance \$6.72 USD

Figura 41. Captura de pantalla de la redirección hacia ambiente Sandbox de PayPal.

Bienvenido, Javier Granada

Prepárate para la película!

Tu compra fue realizada exitosamente. Por favor, acércate al cine 20 min antes de la hora de la función para recibir tu(s) entrada(s)

Hoy en cartelera

Ciudad: Quito

social

x250 px

Figura 42. Redirección final de regreso hacia el sitio web.

Aplicación:

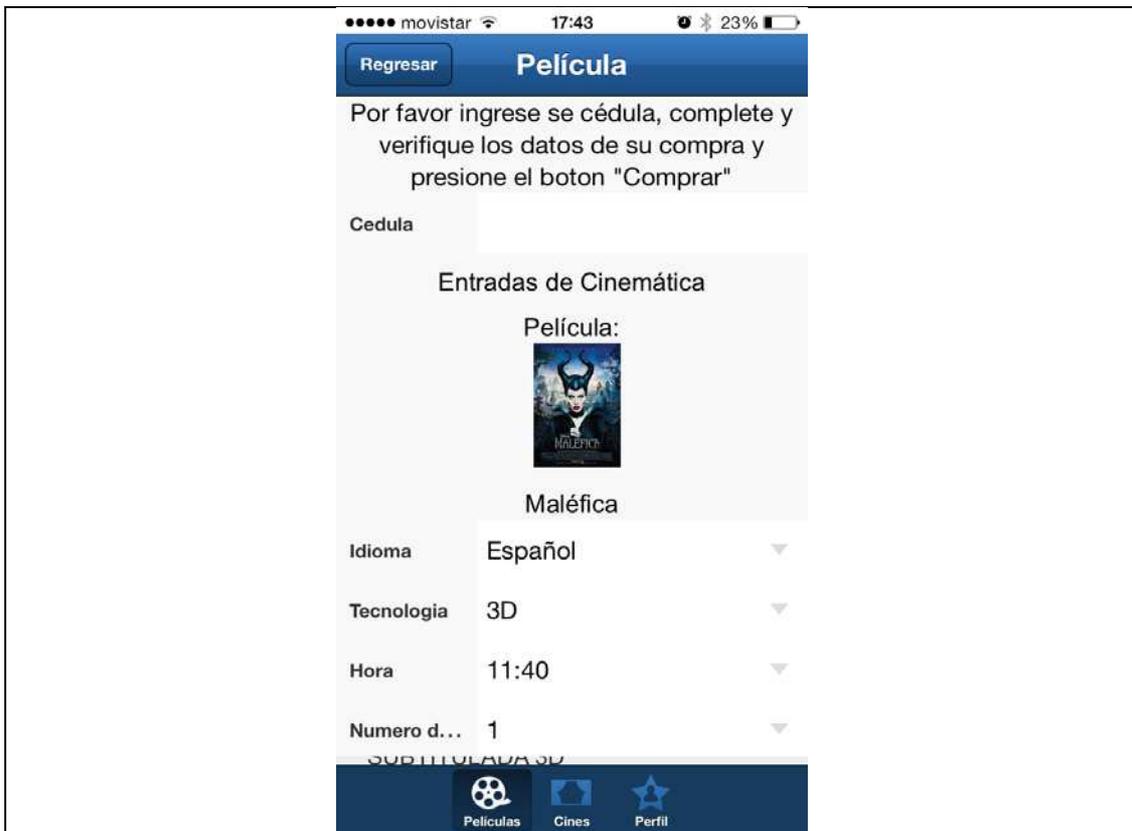


Figura 43. Primer paso para la compra de entradas en la aplicación móvil.

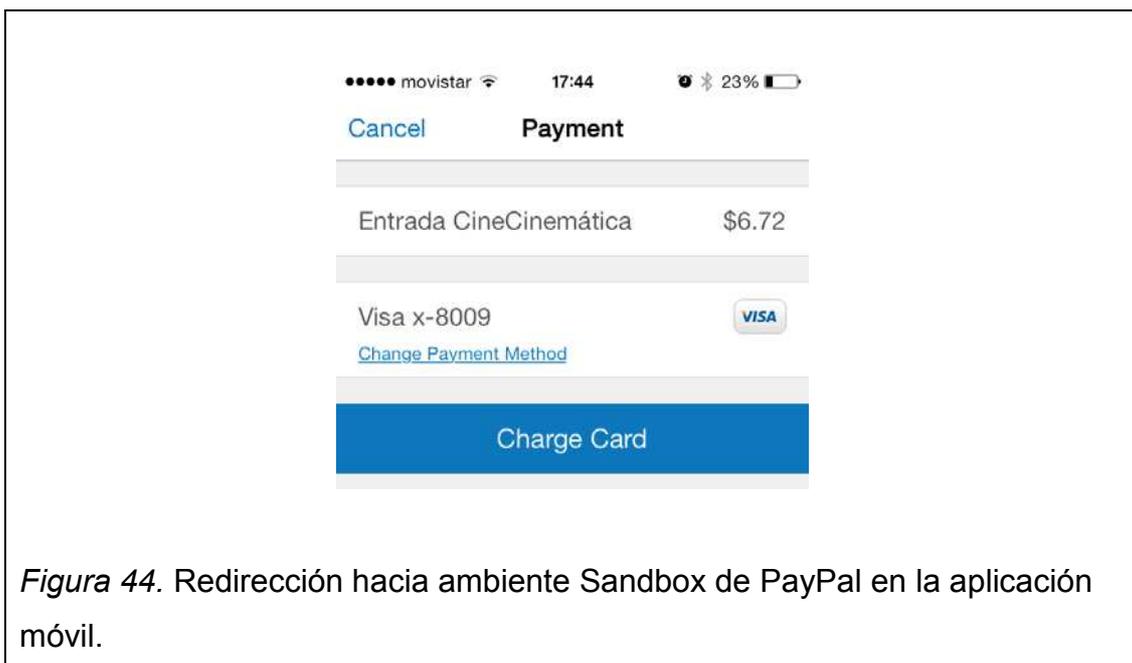


Figura 44. Redirección hacia ambiente Sandbox de PayPal en la aplicación móvil.

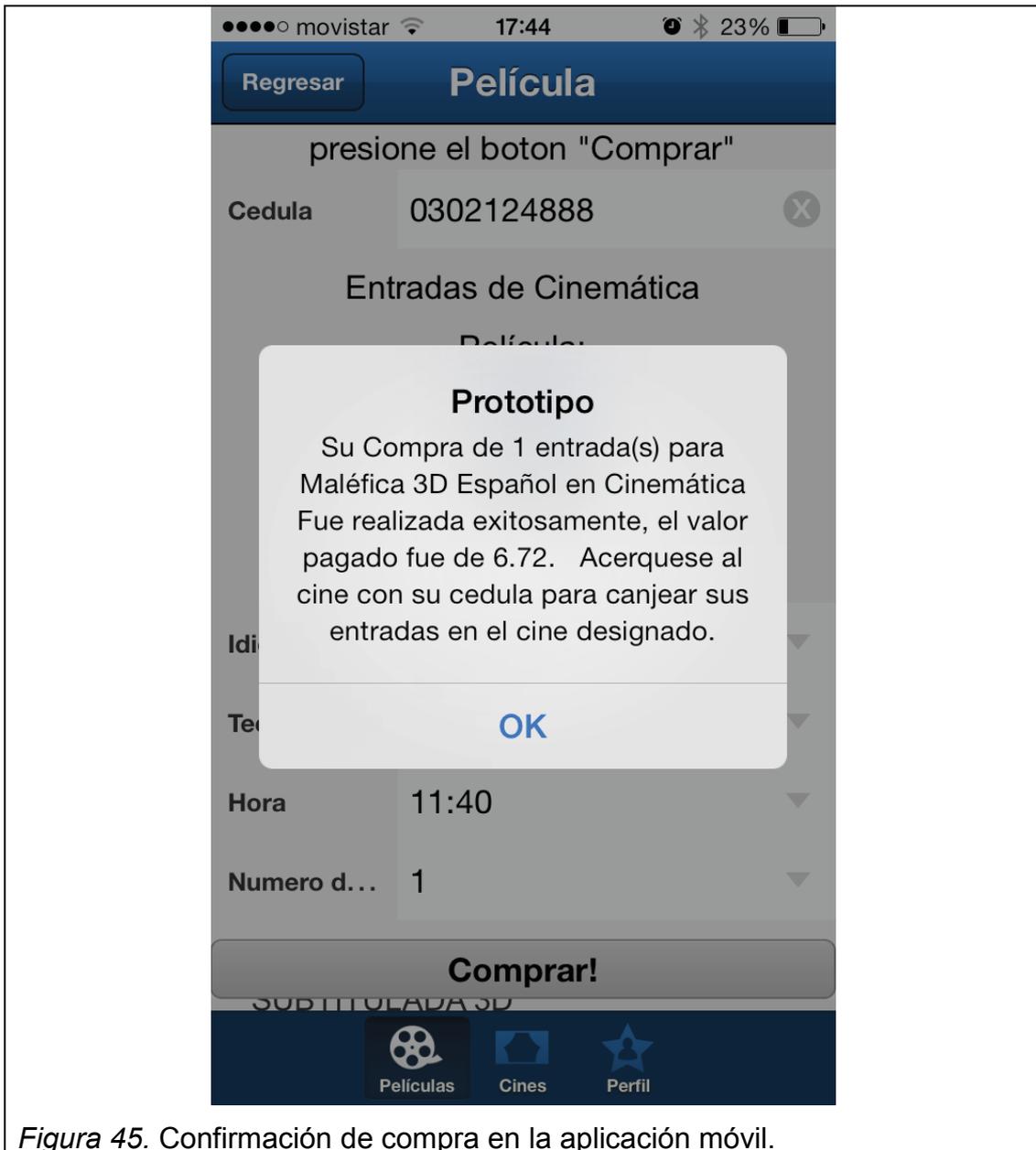


Figura 45. Confirmación de compra en la aplicación móvil.

2.3.16.5. Cierre y Retrospectiva

En el sprint 8 se logró completar el proceso de compra de entradas tanto para la plataforma móvil como web de manera exitosa esto se puede ver en las figuras 40,41,42,43,44,y 45.

Cierta funcionalidad necesaria para el desarrollo del presente sprint se completó en sprint anteriores por lo que la planificación de este requirió menor tiempo y se pudo completar en el tiempo esperado.

3. Implementación y Pruebas

3.1. Implementación

3.1.1. Back-End

Para la implementación del back-end del sistema se utilizará la infraestructura de Amazon “AWS”

3.1.1.1. Instalación de Actualizaciones de Linux, Ajuste de zonas horarias

Para instalar las actualizaciones de Linux utilizaremos “yum” la cual es una utilidad para manejar paquetes desde la línea de comando utilizada en sistemas operativos Linux.

Se utilizan los siguientes procedimientos:

```
sudo yum -y update
```

```
sudo ln -sf /usr/share/zoneinfo/America/Bogota \ /etc/localtime
```

```
sudo yum install -y gcc make
```

Gcc es una colección de compiladores de GNU con el cual podemos compilar C, C++, Objective C, etc

3.1.1.2. Instalación de PHP y extensiones del mismo

Se instalan los paquetes necesarios de PHP y extensiones de desarrollo necesarias para que el código y configuraciones realizadas en PHP funcionen correctamente.

```
sudo yum install -y php
sudo yum install -y php-devel php-mysql php-pdo
sudo yum install -y php-pear php-mbstring
sudo pear install Log
sudo yum install -y pcre-devel
```

3.1.1.3. Instalación de Apache Httpd

Se instala el servidor de http de apache llamado HTTPD

```
sudo yum -y install httpd
sudo service httpd start
```

3.1.1.4. Instalación de Mysql

Se instala paquetes de mysql y se inicia el servicio en el servidor

```
sudo yum -y install mysql-server mysql
sudo service mysqld start
```

3.1.1.5. Instalación de Modulo PHP APC

Existen varias formas de instalar el módulo APC

```
sudo yum -y install httpd-devel  
sudo pecl install apc-beta  
echo "extension=apc.so" | sudo tee /etc/php.d/apc.ini
```

Se utilizó la forma sencilla.

```
sudo yum install -y php-pecl-apc
```

instala todos los paquetes necesarios.

3.1.1.6. Instalación de Modulo Mod-WSGI

Una vez descargado el paquete de mod_wsgi se lo instala usando:

```
sudo apt-get install make
```

A continuación para cargar el modulo en Apache:

```
cd /etc/apache2
```

```
sudo cp httpd.conf httpd.conf.OLD
```

```
sudo vi httpd.conf
```

```
LoadModule wsgi_module modules/mod_wsgi.so
```

```
sudo apachectl restart
```

3.1.1.7. Configuraciones PHP

Se configura php para ocultar código php del mundo y ajustar zona horaria

```
sudo nano /etc/php.ini
```

```
expose_php = Off  
  
memory_limit = 128M  
  
date.timezone = America/Bogota
```

3.1.1.7. Iniciar servicios Apache en el servidor

```
sudo /sbin/chkconfig --levels 235 httpd on  
  
sudo service httpd start
```

3.1.1.8. Configuración de base de datos de producción

La base de datos que se utilizó fue MySQL en su versión 5.5.27. La instancia de la misma se encuentra en el servicio RDS de Amazon y su procesamiento está dado por el servidor EC2 de la misma empresa. Se realizó la migración de la base de datos local a la de producción a través de una herramienta de código abierto llamada south-django que automatiza la implementación de sistemas de bases de datos en aplicaciones basadas en Django.

3.2. Pruebas

Para asegurar el correcto funcionamiento del sistema se probaron todas y cada una de las historias de usuario dentro de la lista de producto. Para mantener este proceso ágil, acorde a la metodología usada para el desarrollo, se realizó un sprint de pruebas en el cual se especificaron todas las tareas a realizar para completar las pruebas exitosamente.

Adicionalmente, con el propósito de brindar un mayor sentido de realidad a la compra de boletos dentro del prototipo, se creó una simulación de la conexión con los sistemas de una cadena de cines para verificar el inventario de los complejos, crear registro de las compras y generar reportes de las mismas.

3.2.1. Sprint de Pruebas

3.2.1.1. Objetivo

Asegurar el cumplimiento de los objetivos de los sprints previos mediante la comprobación y verificación concurrente de todas y cada una de las funcionalidades de cada historia de usuario dentro de estos.

3.2.1.2. Alcance

Este sprint tendrá una duración de una semana, al final de las cuales se verificará la calidad del sistema como un todo asegurando la funcionalidad de cada una de sus partes.

3.2.1.3. Lista de Sprint

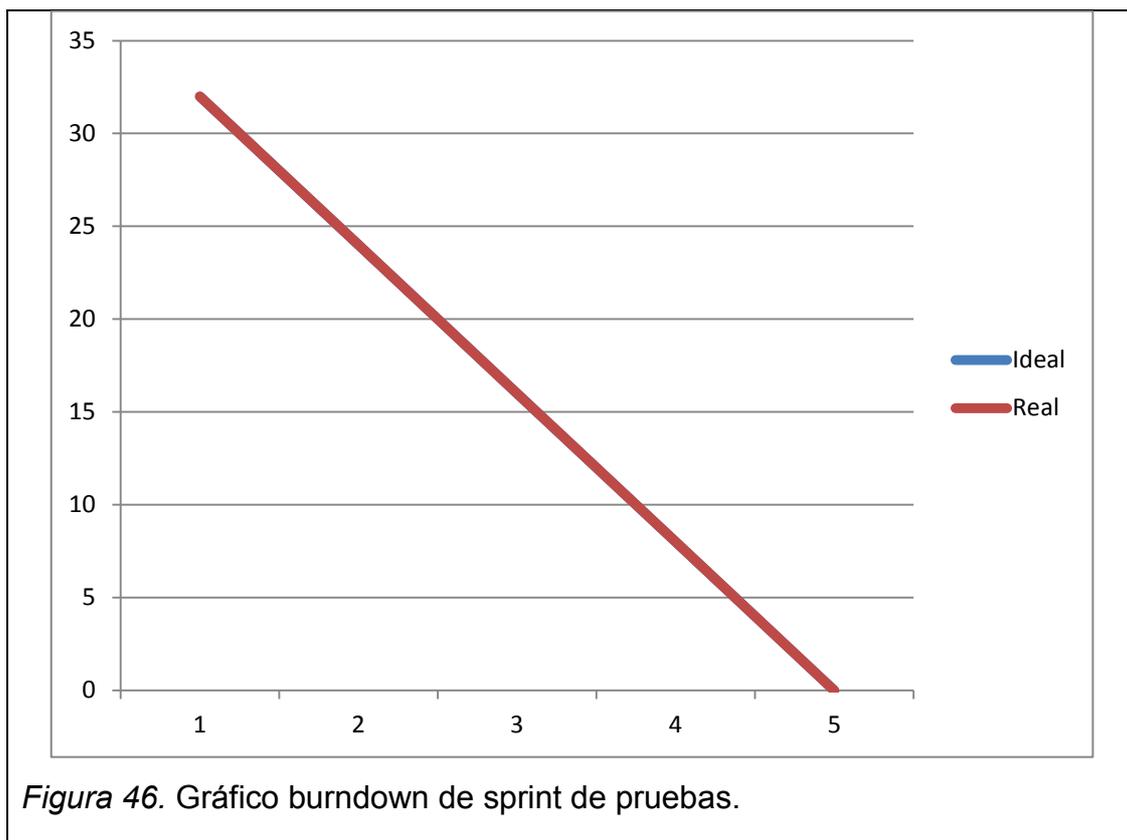
Cada una de las pruebas en la presente tabla se realizaron cinco veces en cada plataforma para asegurar su funcionamiento y corregir posibles errores.

Tabla 17. Lista del sprint de pruebas.

User story	Tarea	Responsable	Plataforma	Estimado (h)	Real (h)	Estado
N/A	Prueba de ingreso al sistema con usuario social			3	3	Hecho
	Subtarea Probar comportamiento de botón (iOS:4 éxitos de 5, Android: 3/5, Web: 4/5)	SM	Móvil	1	1	Hecho
	Subtarea Probar despliegue de ventana de permisos (iOS:5/5, Android: 5/5, Web: 4/5)	JG	Web/Móvil	1	1	Hecho
	Subtarea Probar el regreso a la aplicación y actualización de datos de usuario (iOS:4/5, Android: 4/5, Web: 5/5)	JG	Web/Móvil	1	1	Hecho
N/A	Prueba de ingreso al sistema con usuario anónimo			2	2	Hecho
	Subtarea Probar comportamiento de botón (iOS:5/5, Android: 5/5, Web: 5/5)	SM	Móvil	1	1	Hecho
	Subtarea Probar la redirección a los carruseles de películas (iOS:5/5, Android: 5/5, Web: 5/5)	SM	Móvil	1	1	Hecho
N/A	Prueba de consulta de cartelera por película e información general			4	4	Hecho
	Subtarea Probar comportamiento y comprobar datos con BDD y sitio oficial la consulta para (día actual, día siguiente, dos días después, próximas dos horas y semana actual) (iOS:5/5, Android: 4/5, Web: 5/5)	SM	Web/Móvil	2	2	Hecho
	Subtarea Verificar datos de película y correcta reproducción de tráiler (iOS:5/5, Android: 5/5, Web: 5/5)	JG	Web/Móvil	2	2	Hecho
N/A	Prueba de consulta de cartelera por cine			4	4	Hecho
	Subtarea Probar comportamiento y comprobar datos con BDD y sitio oficial la consulta para (día actual, día siguiente y dos días después) (iOS:5/5, Android: 5/5, Web: 5/5)	JG	Web/Móvil	2	2	Hecho
	Subtarea Verificar datos de dirección y teléfono de cine (iOS:5/5, Android: 5/5, Web: 5/5)	SM	Web/Móvil	2	2	Hecho
N/A	Prueba de calificación de película y consulta de la misma			4	4	Hecho
	Subtarea Probar el comportamiento de plugin de calificaciones y estrellas y ventana para publicación en Facebook (iOS:3/5, Android: 3/5, Web: 4/5)	JG	Web/Móvil	2	2	Hecho

	Subtarea	Comprobar callback, publicación y datos en BDD y módulo de usuario (iOS:3/5, Android: 2/5, Web: 4/5)	SM	Web/Móvil	2	2	Hecho
N/A	Prueba de guardado de película a estrenos esperados consulta				4	4	Hecho
	Subtarea	Probar comportamiento y despliegue de botón, y ventana para publicación en Facebook (iOS:5/5, Android: 4/5, Web: 5/5)	JG	Web/Móvil	2	2	Hecho
	Subtarea	Comprobar callback, publicación y datos en BDD y módulo de usuario (iOS:5/5, Android: 4/5, Web: 3/5)	SM	Web/Móvil	2	2	Hecho
N/A	Prueba de guardado de cine a lista de favoritos y consulta				3	3	Hecho
	Subtarea	Probar comportamiento y despliegue de botón (iOS:5/5, Android: 5/5, Web: 5/5)	JG	Web/Móvil	1	1	Hecho
	Subtarea	Comprobar callback y datos en BDD y módulo de usuario (iOS:3/5, Android: 3/5, Web: 4/5)	SM	Web/Móvil	2	2	Hecho
N/A	Prueba de compra de boleto				8	8	Hecho
	Subtarea	Verificar funcionalidad pantalla de selección de opciones (idioma, tecnología, hora, cantidad) (iOS:5/5, Android: 5/5, Web: 4/5)	SM	Móvil	2	2	Hecho
	Subtarea	Probar funcionalidad de modal de confirmación de compra (iOS:4/5, Android: 4/5, Web: 4/5)	JG	Web	2	2	Hecho
	Subtarea	Probar comportamiento de botón de compra y pantalla de ingreso de datos de pago y confirmación de compra (iOS:5/5, Android: 5/5, Web: 5/5)	SM	Web/Móvil	2	2	Hecho
	Subtarea	Verificación de ingreso de datos en BDD y débito de saldo en cuenta bancaria (iOS:4/5, Android: 3/5, Web: 5/5)	JG	Web/Móvil	2	2	Hecho

Nota: Después de corregir errores principalmente de código se corrieron nuevamente todas las pruebas las mismas que resultaron exitosas y de esta manera se aseguró el correcto funcionamiento del sistema.



3.2.1.5 Cierre y retrospectiva

Se verificaron mediante las pruebas, todas las historias de usuarios y funcionalidades y, a diferencia de los sprints previos, no presenta ningún demo ni una parte funcional del sistema pero representa un paso muy importante para cumplir los objetivos generales de este proyecto.

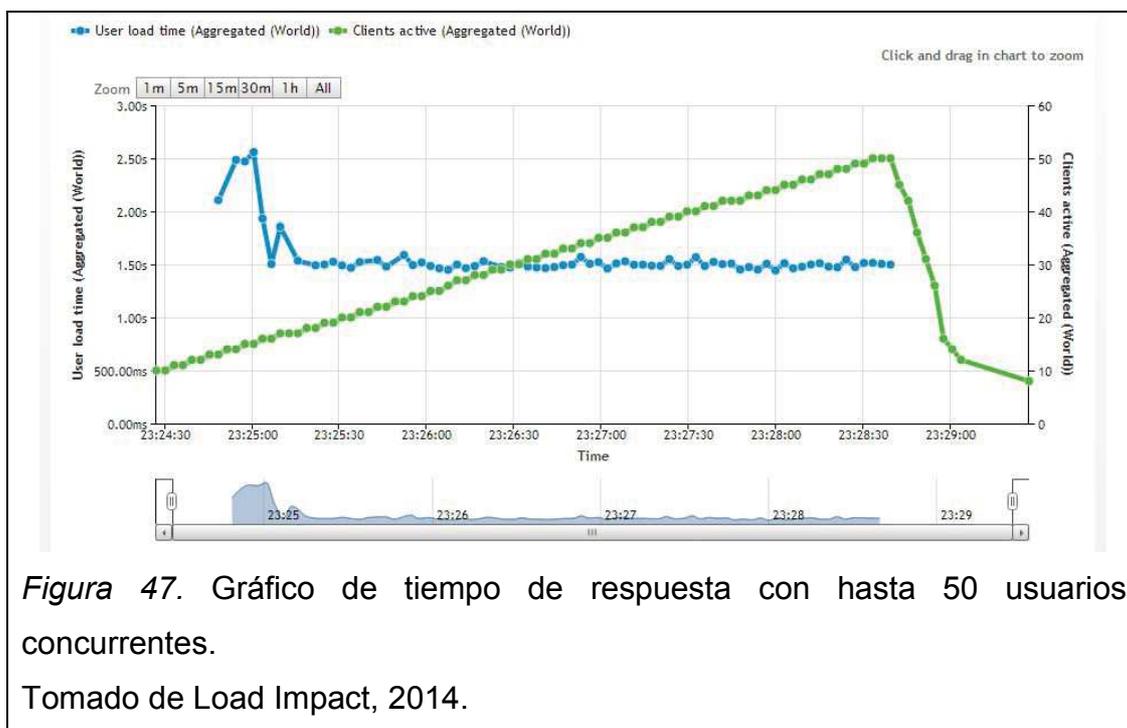
Las pruebas se realizaron sin ningún retraso gracias a tareas de pruebas que se implementaron dentro de cada sprint.

3.2.2. Pruebas no funcionales

3.2.2.1. Prueba de carga

Se realizó una prueba de carga con la herramienta Load Impact, esta herramienta permite realizar pruebas realísticas de usuarios simulados. Esta herramienta simula el tráfico como lo haría un usuario normal por lo que permite comprobar el estado o comportamiento del servidor con varios usuarios concurrentes.

Se realizó la simulación con 50 usuarios concurrentes, los que fueron aumentando linealmente. Esto se puede ver en la figura 47, la cantidad de usuarios concurrentes se representa por la línea de color verde. A su vez los tiempos de carga representados en la figura 47 por la línea de color azul se mantuvieron sin ningún cambio, lo que comprueba que el servidor se encuentra bien configurado y podría soportar un ambiente de producción.



3.2.2.2. Prueba de carga por URLs

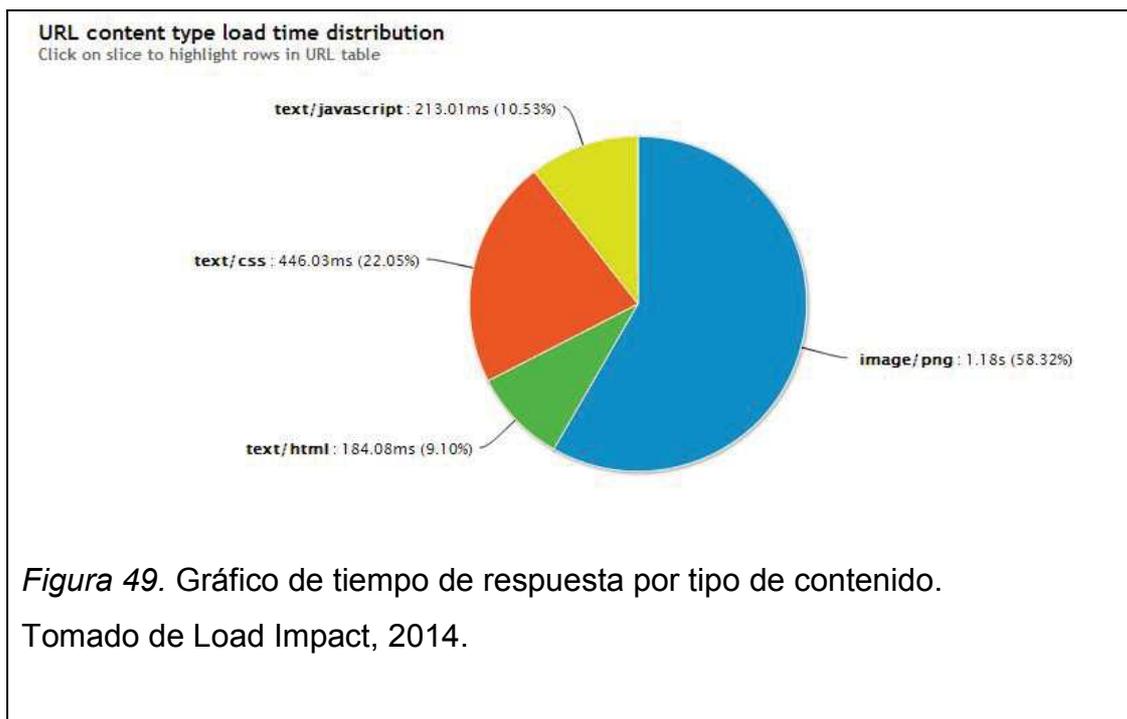
La herramienta Load Impact prueba varios subdominios en la URL y nos devuelve la cantidad de veces que se accedió a cada subdominio con su porcentaje de éxito y tiempos de respuesta. Como se puede ver en la figura 48 todas las pruebas se realizaron exitosamente.

URL	Load zone	User scenario	Successful	Failed	Last avg
http://prototipocines.com/	Aggregated (World)	Auto-generated (8/3/2014 5:21:36 PM)	265	0	184.08ms
http://prototipocines.com/	Dublin, IE (Amazon)	Auto-generated (8/3/2014 5:21:36 PM)	265	0	184.08ms
http://prototipocines.com/ap/css/bootstrap.css	Aggregated (World)	Auto-generated (8/3/2014 5:21:36 PM)	265	0	446.03ms
http://prototipocines.com/ap/css/bootstrap.css	Dublin, IE (Amazon)	Auto-generated (8/3/2014 5:21:36 PM)	265	0	446.03ms
http://prototipocines.com/js/bootstrap.min.js	Aggregated (World)	Auto-generated (8/3/2014 5:21:36 PM)	265	0	257.97ms
http://prototipocines.com/js/bootstrap.min.js	Dublin, IE (Amazon)	Auto-generated (8/3/2014 5:21:36 PM)	265	0	257.97ms

Figura 48. Gráfico de tiempos de respuesta por URLs. Tomado de Load Impact, 2014.

3.2.2.3. Prueba de tiempo de carga por tipos de contenido

Load Impact también prueba el tipo de contenido en el dominio y nos devuelve de manera gráfica, como se puede ver en la figura 49, la distribución en los tiempos de respuesta según el tipo de contenido. De esta manera nos ayuda a ver que es lo que más tiempo toma y como se podría optimizar.



3.2.2.4. Métricas de servidor EC2 y base de datos RDS

El servicio EC2 y RDS de Amazon nos permiten obtener métricas de uso de componentes claves del servidor como el uso del CPU o la cantidad de conexiones a la base de datos. Mediante estas métricas se pudo observar el comportamiento de ambos servidores durante la prueba de carga, y se pudo ver en la figura 50, que el aumento de procesamiento en ambos servidores es mínimo llegando al 4% en el caso del servidor RDS y al 2.8 % en el caso de EC2.

Esto prueba que ambos servidores se encuentran preparados para un ambiente de producción.

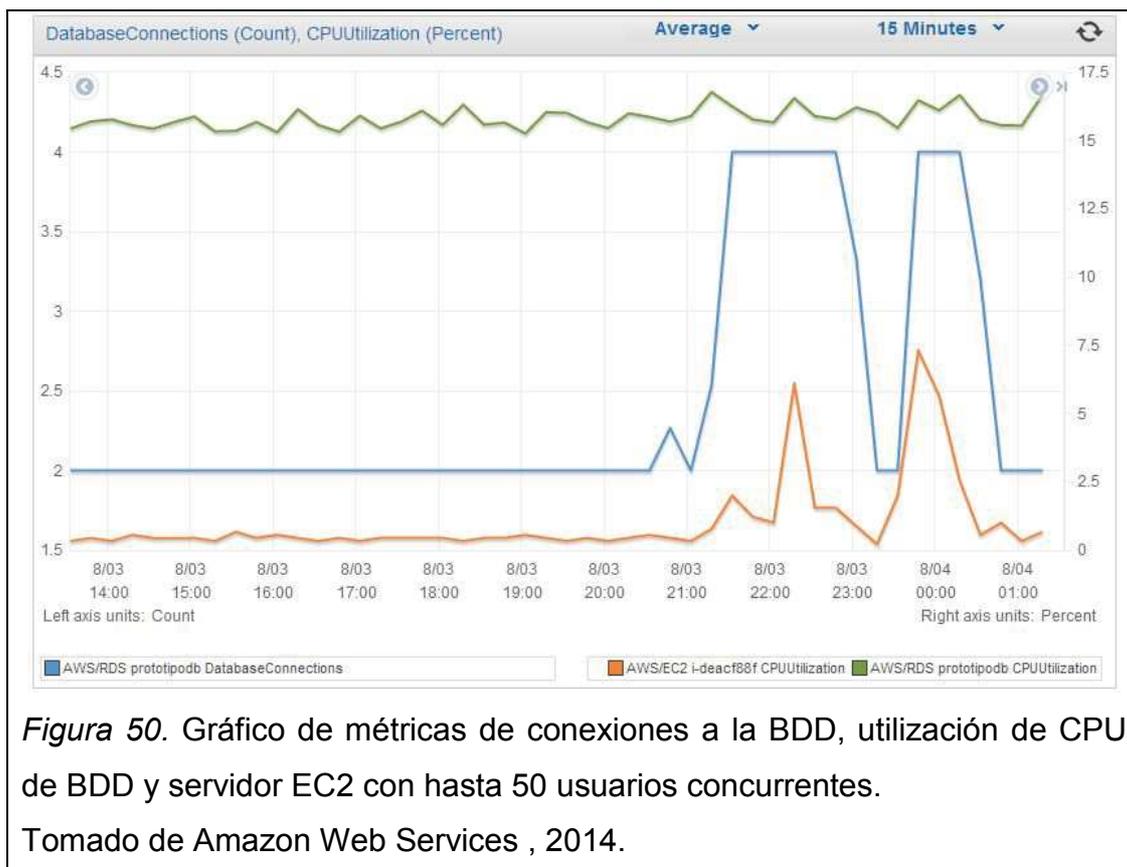


Figura 50. Gráfico de métricas de conexiones a la BDD, utilización de CPU de BDD y servidor EC2 con hasta 50 usuarios concurrentes.

Tomado de Amazon Web Services , 2014.

4. Conclusiones y Recomendaciones

4.1. Conclusiones

Se desarrolló el sistema para todas las plataformas definidas en el proyecto (iOS, Android y Web) dentro del marco de tiempo establecido, satisfaciendo las necesidades identificadas en los antecedentes, asegurando el cumplimiento de todas las funcionalidades y especificaciones que se expusieron en la visión y el alcance.

Para levantamiento de los requerimientos funcionales del prototipo del sistema se utilizó historias de usuario, las mismas ayudan a plasmar las necesidades

de una manera natural sin tener que llenar formularios complejos o realizar diagramas técnicos que entorpecen el proceso y causan rechazo en los usuarios.

El uso del framework Sencha, una herramienta especializada en el diseño de interfaces de usuario para plataformas móviles y HTML5 en web, permitió utilizar elementos gráficos actuales, como los carruseles y posters de películas que se asemejan a los flujos de navegación de sitios como Netflix o Hulu, empresas a la vanguardia en la industria del entretenimiento mediante la transmisión (streaming) bajo demanda de películas y video, brindando así, una experiencia de uso familiar y reduciendo la curva de aprendizaje para el uso del prototipo.

La integración de las diferentes arquitecturas, lenguajes, plataformas y el uso de herramientas de desarrollo híbridas logró una disminución aproximada del 60% en el tiempo de desarrollo de la aplicación móvil ya que se utilizó un único código fuente base, invirtiendo aproximadamente 26 horas en el empaquetamiento final para iOS y Android.

Adicionalmente, al ser una aplicación híbrida, se reduce aproximadamente un 25% el tiempo que toman tareas como el mantenimiento y la actualización del código, así como la inclusión de nuevas plataformas.

El uso de Django como framework, con su principio DRY (Don't Repeat Yourself), permitió que el código se mantenga siempre optimizado, adicionalmente, y, como Django maneja automáticamente el modelamiento de la base de datos, se ahorró aproximadamente un 40% en la puesta en producción de la base de datos.

Se estandarizó y esquematizó el proceso de unificación de información recibida por los cines para que el mismo sea escalable y administrable de tal manera que un nuevo complejo de cines sea fácilmente integrado en el sistema.

La integración con Facebook fue una parte muy importante de la experiencia de uso del sistema, se logró generar un valor agregado tanto para el proyecto como para el usuario ya que facilita el registro y al mismo tiempo permite acceder a más funcionalidades y opciones de comunicación con su red social. Durante el proceso de pruebas, se realizaron varias simulaciones de registro y uso, generando publicaciones reales en Facebook desde la aplicación móvil y el sitio web.

Gracias al uso de SCRUM en la fase de desarrollo se liberaron versiones funcionales e incrementales del producto cada dos semanas, permitiendo así, mantener un proceso de desarrollo ágil e iterativo, generando valor agregado para los usuarios quienes se familiarizan con el producto desde fases tempranas.

El incluir sprints adicionales para configuraciones y pruebas, tanto al inicio como al final del desarrollo, fue de gran ayuda para desarrollar los requisitos no funcionales esenciales para el éxito del proyecto y asegurar el correcto funcionamiento del sistema, gracias a esto, durante el primer sprint las tareas de configuración tomaron cerca del 5% del mismo en contraste a un estimado del 60% en caso de no incluir dichos sprints.

4.2. Recomendaciones

Durante el desarrollo de este sistema, se presentaron problemas y dificultades que todo proyecto de este tipo afronta, gracias a esta experiencia, se

generaron las siguientes recomendaciones para futuros proyectos de esta índole.

La inclusión de una persona especialista en diseño gráfico e ilustración como parte del equipo para enriquecer la estética, colores y distribución de elementos. Debido a que esta no es un área de especialización de los desarrolladores involucrados, el desarrollo gráfico demandó un tiempo excesivo aproximado de 32 horas.

Se recomienda un análisis más profundo de los índices en la base de datos debido a que Python-Django consume recursos dentro del servidor de BDD. De esta manera se podrá mejorar los tiempos de respuesta en escritura y principalmente en lectura de la base de datos.

Una parte importante de un sistema como el presente es la integración de la información proporcionada por las cadenas de cine, por lo que se recomienda que las cadenas desarrollen una herramienta automatizada como lo es un API (Application Programming Interface) para el acceso a la información necesaria.

Un sistema de versionamiento en la nube ofrece varias ventajas, principalmente facilita el desarrollo colaborativo, la unificación de código, y respaldos del mismo, por lo tanto, es altamente recomendable el uso del mismo. Por otro lado, la instalación de un sistema de este tipo dentro del servidor de producción, permite disminuir el tiempo de implementación en aproximadamente un 80% y agilizar el flujo de actualización de cambios entre ambientes de desarrollo y producción.

5. Referencias:

Adobe Systems Inc. (2013a). *Adobe Announces Agreement to Acquire Nitobi, Creator of PhoneGap*. Recuperado el 28 de noviembre de 2013 de <http://www.adobe.com/aboutadobe/pressroom/pressreleases/201110/AdobeAcquiresNitobi.html>

Adobe Systems Inc. (2013b). PhoneGap, Cordova, and what's in a name? Recuperado el 28 de noviembre de 13 de <http://phonegap.com/2012/03/19/phonegap-cordova-and-what%E2%80%99s-in-a-name/>

Adobe Systems Inc. (2013c). Recuperado el 28 de noviembre de 13 de <http://phonegap.com/>

Amazon. (2013a). Amazon EC2. Recuperado 28 de noviembre de 2013 de <http://aws.amazon.com/es/ec2/>

Amazon. (2013b). Amazon RDS. Recuperado 28 de noviembre de 2013 de <http://aws.amazon.com/es/rds/>

Amazon. (2013c). Amazon S3. Recuperado 28 de noviembre de 2013 de <http://aws.amazon.com/es/s3/>

Amazon. (2013d). AMI de Amazon Linux. Recuperado el 28 de noviembre de 2013 de <http://aws.amazon.com/es/amazon-linux-ami/>

Amazon Web Services. (2014). AWS Recuperado el 10 de mayo de 2014 de <http://aws.amazon.com/>

Apache Software Foundation. (2014). About Apache. Recuperado el 21 de Enero de 2014 de http://httpd.apache.org/ABOUT_APACHE.html

- Apple. (2013). Programming with Objective-C. Recuperado el 27 de noviembre de 2013 de <https://developer.apple.com/library/mac/documentation/cocoa/conceptual/ProgrammingWithObjectiveC/Introduction/Introduction.html>
- Compass. (2013). Getting Started with Compass. Recuperado el 27 de noviembre de 2013 de <http://compass-style.org/help/>
- Catlin, H., Weizenbaum, N. y Eppstein, C. (2013). SASS. Recuperado el 27 de noviembre de 2013 de <http://sass-lang.com/>
- Holovaty Adrian y Kaplan-Moss Jacob. (2009). The Django Book. Recuperado el 27 de noviembre de 2013 de <http://www.djangobook.com/en/2.0/index.html>
- Intel. (2012). The Development of Mobile Applications using HTML5 and PhoneGap on Intel Architecture-Based Platforms. Recuperado el 6 de enero de 2014 de <http://software.intel.com/en-us/articles/the-development>.
- Jeff Croft. (2007). MyTARDIS 1.99 documentation. Recuperado el 27 de noviembre de 2013 de <http://pythonhosted.org/MyTARDIS/architecture.html>
- Load Impact. (2014). Recuperado el 20 de Mayo de 2014 de <http://loadimpact.com/>
- Microsoft Developer Network. (2014). Recuperado el 10 de abril de 2014 de <http://msdn.microsoft.com/es-es/library/jj161047.aspx>
- Mozilla Developer Network. (s.f). Recuperado el 27 de noviembre de 2013 de https://developer.mozilla.org/en-US/docs/Web/JavaScript/About_JavaScript
- Oracle. (2013). Java. Recuperado el 27 de noviembre de 2013 de <http://www.java.com/>

Python Organization. (s.f). *About Python*. Recuperado el 28 de noviembre de 2013 de <http://www.python.org/about/>.

Scrum Alliance. (s.f). Recuperado el 27 de noviembre de 2013 de <http://www.scrumalliance.org/>

Kniberg Henrik (2007). *Scrum y XP desde las Trincheras*. Estados Unidos de America. C4Media Inc.

Schwaber Ken y Sutherland Jeff. (2013). *La Guía Definitiva de Scrum: Las reglas del Juego*. Recuperado el 27 de noviembre de 2013 de <https://www.scrum.org/Portals/0/Documents/Scrum%20Guides/2013/Scrum-Guide-ES.pdf#zoom=100>

Sencha. (2013). Recuperado el 28 de noviembre de 13 de <http://www.sencha.com/products/touch/>

The PHP Group. (2014). APC Introducción Recuperado el 6 de febrero de 2014 de <http://cl1.php.net/manual/es/intro.apc.php>

WSGI.org. (s.f). What is WSGI? Recuperado el 28 de noviembre de 2013 de <http://wsgi.readthedocs.org/en/latest/what.html>

Word Wide Web Consortium. (s.f). Guía Breve de CSS. Recuperado el 24 de noviembre de 2013 de <http://www.w3c.es/Divulgacion/GuiasBreves/HojasEstilo>.

W3Schools. (2013). HTML 5. Recuperado el 27 de Noviembre de 2013 de http://www.w3schools.com/html/html5_intro.asp

Sencha. (2013a). Sencha touch features. Tomado el 10 de diciembre de 2013 de <http://www.sencha.com/products/touch/features/>