



UNIVERSIDAD DEL BÍO-BÍO, CHILE

FACULTAD DE CIENCIAS EMPRESARIALES

SISTEMA DE PRONÓSTICO DEL TIEMPO EN BASE A MODELO  
ATMOSFÉRICO WRF IMPLEMENTADO POR EL  
DEPARTAMENTO DE GEOFÍSICA DE LA UNIVERSIDAD DE  
CONCEPCIÓN MEDIANTE DISPOSITIVOS MÓVILES PARA LA  
REGIÓN DEL BIOBÍO

PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN INFORMÁTICA

ALUMNOS: PAOLA TORRES FERRADA Y RENATO ÁVILA MOMBERG

PROFESOR GUÍA: MANUEL CRISOSTO MUÑOZ

PROFESOR COGUÍA: OSVALDO ARTAL ARRIETA

2014

Escuela de Ingeniería Civil en Informática  
Departamento de Sistemas de Información

# Agradecimientos

Nuestras primeras palabras de agradecimientos son dirigidas a Dios, por acompañarnos, darnos salud y fortaleza suficiente a lo largo de estos cinco años.

Agradecemos a nuestras familias correspondientes de quienes siempre hemos recibido su amor, comprensión y paciencia, además de su incondicionalidad en los buenos y malos momentos, que nos ayudó a lograr nuestras metas. Por ser un pilar fundamental en nuestras vidas.

A todos los profesores quienes siempre estuvieron allí, brindándonos siempre su orientación y ayuda en todo momento. Pero en especial a nuestro profesor guía, Manuel Crisosto Muñoz, gracias por creer en nosotros, sus valiosos consejos y conocimientos fueron un aporte invaluable en nuestra formación.

Y a nuestro profesor coguía Osvaldo Artal Arrieta por su paciencia y compromiso a lo largo del desarrollo de este proyecto.

Adicionalmente queremos agradecer a nuestro grupo de amigos: Jorge Elgueta Morales, Heber Gálvez Ojeda, Alejandro Neira San Martín, Joel Torres Carrasco, Cristian Vallejos Vega, por todo la ayuda prestada y palabras de aliento en estos años.

Gracias a todos.

# Resumen

Este proyecto se presenta para dar conformidad a los requisitos exigidos por la Universidad del Bío-Bío en el proceso de titulación para la carrera de Ingeniería Civil en Informática. El proyecto titulado “Sistema de pronóstico del tiempo en base a modelo atmosférico WRF implementado por el departamento de Geofísica de la Universidad de Concepción mediante dispositivos móviles para la región del Biobío”, cuyo propósito es la creación de una aplicación móvil para dispositivos Android que presente el pronóstico fiable del tiempo utilizando la información generada por el modelo de predicción climático WRF el cual es implementado por el departamento de Geofísica de la Universidad de Concepción con quienes se trabaja en conjunto.

El departamento de Geofísica busca darle un buen uso a la información generada por el modelo WRF, información que en su gran mayoría no está siendo aprovechada, es por esto que se desea proyectar dicha información para ser un aporte a la comunidad regional.

Para el desarrollo de la aplicación móvil, se hace uso de características nativas del dispositivo como lo es el sensor GPS para encontrar la posición del usuario y brindar la opción de un pronóstico del día o uno resumido con información para tres días. También, es posible hacer uso del mapa que posee la aplicación, mapa que es provisto por Google maps, para poder de manera interactiva seleccionar alguna comuna de la cual se desee consultar el pronóstico del tiempo.

En términos de programación no solo se trabaja con la tecnología relativa al sistema operativo Android, sino que también se utiliza PHP para el servidor y matlab para la interacción con el modelo WRF.

# Abstract

This project appears to provide pursuant to the requirements of the University of Bío-Bío in the titling process for Civil Engineering in Computer Science. The project entitled “ forecast system based on WRF atmospheric model implemented by the Department of Geophysics of the University of Concepción through mobile devices for the Biobío region”, whose purpose is to create a mobile application for Android devices presenting reliable forecast using information generated by the climate prediction model WRF which is implemented by the Department of Geophysics of the University of Concepcion, with who we are working.

Geophysics department is looking for make good use of the information generated by the WRF model, information that is mostly not being utilized, that is why they want to project this information to be a contribution to the regional community.

For the development of the mobile application, using native device features such as the GPS sensor to find the position of the user and provide the option of a day or a forecast summary information for three days . Also, it is possible to make use of the map that has the application, map provided by Google maps, to interactively select a commune to view the weather.

In programming terms not only works with the technology of Android operating system, but PHP is also used for the server and matlab for interaction with the WRF model.

# Índice general

<b>Agradecimientos</b>	<b>II</b>
<b>Resumen</b>	<b>III</b>
<b>Abstract</b>	<b>IV</b>
<b>Introducción</b>	<b>1</b>
<b>1. Definición de la institución</b>	<b>4</b>
1.1. Descripción de la institución . . . . .	4
1.1.1. Antecedentes generales de la Institución . . . . .	4
1.1.2. Entorno . . . . .	4
1.1.3. Misión . . . . .	4
1.1.4. Visión . . . . .	5
1.1.5. Organigrama . . . . .	5
1.2. Descripción del área de estudio . . . . .	8
1.3. Descripción de la problemática . . . . .	8
1.3.1. Introducción . . . . .	8
1.3.2. La problemática . . . . .	12
1.3.3. Solución . . . . .	12
<b>2. Estado del arte</b>	<b>13</b>
2.1. Introducción . . . . .	13
2.2. Dispositivos móviles inteligentes . . . . .	13
2.2.1. Plataformas disponibles . . . . .	13
2.2.2. Justificación plataforma escogida . . . . .	16
2.3. Sistema operativo Android . . . . .	17
	v

2.3.1.	Introducción a Android . . . . .	17
2.3.2.	Versiones de Android . . . . .	17
2.3.3.	Componentes fundamentales . . . . .	19
	<b>Entorno de desarrollo . . . . .</b>	<b>19</b>
	<b>Estructura de un proyecto en Android . . . . .</b>	<b>24</b>
	<b>Componentes de una aplicación Android . . . . .</b>	<b>30</b>
2.4.	WRF (Weather Research and Forecasting Model) . . . . .	31
2.4.1.	Introducción . . . . .	31
2.4.2.	Funciones del WPS (WRF Preprocessing System) . . . . .	32
	Programa Geogrid . . . . .	33
	Programa Ungrib . . . . .	33
	Programa Metgrid . . . . .	33
2.4.3.	Creación de un dominio de estudio para WRF . . . . .	34
2.5.	MATLAB (MATrix LABoratory) . . . . .	41
2.5.1.	Introducción . . . . .	41
2.5.2.	Características Principales . . . . .	41
2.6.	Aplicaciones móviles similares . . . . .	42
<b>3.</b>	<b>Definición Proyecto . . . . .</b>	<b>47</b>
3.1.	Justificación Proyecto . . . . .	47
3.2.	Objetivos del proyecto . . . . .	47
3.3.	Ambiente de ingeniería de software . . . . .	48
3.3.1.	Metodología de desarrollo . . . . .	48
3.3.2.	Técnicas y notaciones . . . . .	49
3.3.3.	Estándares de documentación . . . . .	49
3.3.4.	Herramientas de apoyo al desarrollo . . . . .	49
3.4.	Definiciones, siglas y abreviaciones . . . . .	50
<b>4.</b>	<b>Especificación de requerimientos de software . . . . .</b>	<b>53</b>
4.1.	Alcances . . . . .	53
4.2.	Objetivos del software . . . . .	54
4.2.1.	Objetivos Generales . . . . .	54
4.2.2.	Objetivos Específicos . . . . .	54
4.3.	Descripción global del producto . . . . .	54
4.3.1.	Interfaz de Usuario . . . . .	54
4.3.2.	Interfaz de Hardware . . . . .	55

4.3.3.	Interfaz de Software . . . . .	55
4.3.4.	Interfaz de Comunicación . . . . .	55
4.4.	Requerimientos específicos . . . . .	56
4.4.1.	Requerimientos funcionales del sistema . . . . .	56
4.4.2.	Interfaces externas de entrada . . . . .	57
4.4.3.	Interfaces externas de salida . . . . .	57
4.4.4.	Atributos del producto . . . . .	57
<b>5.</b>	<b>Estudio de factibilidad</b>	<b>61</b>
5.1.	Introducción . . . . .	61
5.2.	Factibilidad técnica . . . . .	61
5.2.1.	Software utilizado para el desarrollo . . . . .	62
5.2.2.	Hardware mínimo para el desarrollo . . . . .	62
5.2.3.	Software mínimo para el funcionamiento . . . . .	63
5.2.4.	Hardware mínimo para el funcionamiento . . . . .	63
5.2.5.	Conocimientos del equipo de trabajo . . . . .	64
5.2.6.	Conclusión de factibilidad técnica . . . . .	65
5.3.	Factibilidad operativa . . . . .	65
5.3.1.	Impactos positivos . . . . .	65
5.3.2.	Impactos negativos . . . . .	65
5.3.3.	Conclusiones de factibilidad operativa . . . . .	66
5.4.	Factibilidad económica . . . . .	66
5.4.1.	Inversión de tangibles . . . . .	66
5.4.2.	Inversión de intangibles . . . . .	66
5.4.3.	Inversión de recursos humanos . . . . .	67
5.4.4.	Conclusión de factibilidad económica . . . . .	67
5.5.	Conclusión del estudio de factibilidad . . . . .	67
<b>6.</b>	<b>Análisis</b>	<b>68</b>
6.1.	Diagrama Caso de Uso . . . . .	68
6.1.1.	Actores . . . . .	68
6.1.2.	Casos de Uso . . . . .	70
6.1.3.	Descripción general del uso del sistema . . . . .	71
6.1.4.	Especificación de los casos de uso . . . . .	71

<b>7. Diseño</b>	<b>81</b>
7.1. Introducción . . . . .	81
7.2. Diseño de arquitectura funcional . . . . .	82
7.3. Diseño de interfaz y navegación . . . . .	86
7.3.1. iconografía . . . . .	86
7.4. Especificación de módulos . . . . .	92
<b>8. Pruebas</b>	<b>95</b>
8.1. Introducción . . . . .	95
8.2. Elementos de pruebas . . . . .	95
8.3. Especificación de pruebas . . . . .	96
8.4. Responsables de las pruebas . . . . .	98
8.5. Calendario de pruebas . . . . .	98
8.6. Detalle de pruebas . . . . .	99
8.7. Conclusión de pruebas . . . . .	99
<b>9. Plan de Implementación y puesta en marcha</b>	<b>100</b>
<b>10. Resumen de esfuerzo requerido</b>	<b>101</b>
<b>11. Conclusiones</b>	<b>103</b>
<b>Referencias</b>	<b>105</b>
<b>1. Planificación inicial del proyecto</b>	<b>109</b>
1.1. Estimación inicial de tamaño . . . . .	111
1.1.1. Clasificación de los actores . . . . .	111
1.1.2. Clasificación de los casos de uso . . . . .	112
1.1.3. Puntos de caso de uso sin ajustar . . . . .	112
1.1.4. Factores técnicos . . . . .	113
1.1.5. Factores ambientales . . . . .	114
1.1.6. Puntos de caso de uso ajustados . . . . .	115
1.1.7. Calculo de esfuerzo requerido . . . . .	115
1.1.8. Contabilización final del tamaño del software . . . . .	116
<b>2. Encuesta a potenciales usuarios</b>	<b>117</b>
2.1. Introducción . . . . .	117

---

2.2. Datos de la encuesta . . . . .	117
2.3. Análisis de respuestas . . . . .	118
2.4. Sugerencias y comentarios . . . . .	124
<b>3. Especificación de pruebas</b>	<b>126</b>
3.1. Prueba bloque 1 . . . . .	126
3.2. Prueba bloque 2 . . . . .	127
3.3. Prueba bloque 3 . . . . .	128
3.4. Prueba bloque 4 . . . . .	129

# Índice de figuras

1.1. Organigrama Corporación UdeC. . . . .	6
1.2. Organigrama Universidad de Concepción. . . . .	7
1.3. Variables de viento. . . . .	9
1.4. Precipitaciones. . . . .	10
1.5. Temperatura y viento. . . . .	11
1.6. Temperatura y humedad. . . . .	11
2.1. Cuota de Mercado global de plataformas de smartphone. . . . .	16
2.2. Porcentaje de utilización de las versiones existentes en Android. . . . .	19
2.3. Menu del SDK Manager para escoger la plataforma Android a utilizar. . . . .	20
2.4. Android Virtual Devices. . . . .	21
2.5. Opciones para editar el emulador en Android. . . . .	23
2.6. Formulario 1. . . . .	25
2.7. Formulario 2. . . . .	26
2.8. Formulario 3. . . . .	27
2.9. Formulario 4. . . . .	28
2.10. Formulario 5. . . . .	29
2.11. Estructura del proyecto en Android. . . . .	30
2.12. Esquema que muestra el flujo de datos y componentes del programa WPS y como los datos llegan finalmente al WRF. Las letras en los rectángulos indican los nombres de los programas que ayudan en el proceso. . . . .	33
2.13. Configuración del dominio. . . . .	34
2.14. Opción de dominio. . . . .	35
2.15. Definición de nombre y descripción del nuevo dominio. . . . .	36
2.16. Definición del dominio. . . . .	37
2.17. Descripción del archivo “namelist.input”. . . . .	38

2.18. Pantalla de ejecución de Preprocesadores. . . . .	39
2.19. Archivos NetCDF creados. . . . .	40
2.20. WeatherPro. . . . .	42
2.21. Go Weather Forecast & Widgets. . . . .	43
2.22. MeteoEarth. . . . .	44
2.23. AccuWeather. . . . .	45
2.24. eWeatherHD & Terremotos. . . . .	46
6.1. Diagrama casos de uso. . . . .	70
7.1. Diseño de arquitectura funcional - nivel general. . . . .	82
7.2. Descomposición funcional - Gestión de modelo WRF. . . . .	83
7.3. Descomposición funcional - Gestión solicitud pronóstico. . . . .	84
7.4. Descomposición funcional - Gestión de mapas. . . . .	85
7.5. Logo de la aplicación. . . . .	86
7.6. íconos para representar pronóstico. . . . .	87
7.7. figura de temperatura. . . . .	88
7.8. ícono de mapa. . . . .	88
7.9. ícono de ubicación. . . . .	88
7.10. ícono de menú. . . . .	88
7.11. Otras figuras. . . . .	88
7.12. Barra de la aplicación android. . . . .	89
7.13. tipos de vistas del mapa. . . . .	90
7.14. Formato para pronóstico de un día. . . . .	91
7.15. Formato para pronóstico de más días. . . . .	91
1.1. Planificación inicial. . . . .	110
2.1. Pregunta 1. . . . .	118
2.2. Pregunta 2. . . . .	118
2.3. Pregunta 3. . . . .	119
2.4. Pregunta 4. . . . .	120
2.5. Pregunta 5. . . . .	121
2.6. Pregunta 6. . . . .	121
2.7. Pregunta 7. . . . .	122
2.8. Pregunta 8. . . . .	122
2.9. Pregunta 9. . . . .	123

---

2.10. Pregunta 10. . . . .	124
----------------------------	-----

# Índice de tablas

2.1. Comparación entre sistemas operativos móviles. . . . .	15
2.2. Tabla con version es y N° de API. . . . .	18
4.1. Requerimientos funcionales. . . . .	56
4.2. Interfaces externas de entrada. . . . .	57
4.3. Interfaces externas de salida. . . . .	57
5.1. Software para el desarrollo. . . . .	62
5.2. Software mínimo para el usuario. . . . .	63
5.3. Software mínimo para el Servidor. . . . .	63
5.4. Suscripción a Google Play . . . . .	67
6.1. Flujo de eventos básicos: <Consultar Posición>. . . . .	71
6.2. Flujo de enentos alternativos: <Consultar Posición>. . . . .	72
6.3. Flujo de eventos básicos: <Datos Modelo>. . . . .	72
6.4. Flujo de eventos básicos: <Calcular Información>. . . . .	73
6.5. Flujo de eventos básicos: <Consultar Pronóstico>. . . . .	74
6.6. Flujo de eventos básicos: <Gestión Solicitud>. . . . .	75
6.7. Flujo de eventos básicos: <Mostrar Información>. . . . .	76
6.8. Flujo de eventos básicos: <Seleccionar Comuna>. . . . .	77
6.9. Flujo de eventos básicos: <Visualizar Mapa>. . . . .	78
6.10. Flujo de eventos básicos: <Actualizar Mapa>. . . . .	79
6.11. Flujo de eventos básicos: <Cambiar Vista Del Mapa>. . . . .	80
6.12. Flujo de eventos básicos: <Consultar Instrucciones>. . . . .	80
7.1. Especificación de módulos - Extraer datos. . . . .	92
7.2. Especificación de módulos - Realizar solicitud. . . . .	92

---

7.3. Especificación de módulos - Calcular datos. . . . .	92
7.4. Especificación de módulos - Recibir datos. . . . .	93
7.5. Especificación de módulos - Desplegar información. . . . .	93
7.6. Especificación de módulos - Obtener posición actual. . . . .	93
7.7. Especificación de módulos - Mostrar mapa. . . . .	94
7.8. Especificación de módulos - Seleccionar comuna. . . . .	94
7.9. Especificación de módulos - Cambiar vista. . . . .	94
8.1. Especificación de las pruebas - parte 1. . . . .	96
8.2. Especificación de las pruebas - parte 2. . . . .	97
8.3. Responsables de las pruebas a realizar. . . . .	98
8.4. Calendario de las pruebas a realizar. . . . .	98
10.1. Resumen de esfuerzo requerido de Paola Torres Ferrada. . . . .	101
10.2. Resumen de esfuerzo requerido de Renato Ávila Momberg. . . . .	102
10.3. Total de esfuerzo requerido. . . . .	102
1.1. Planificación inicial. . . . .	109
1.2. Clasificación de los actores. . . . .	111
1.3. Clasificación de los casos de uso. . . . .	112
1.4. Escala de evaluación para factores técnicos y ambientales. . . . .	113
1.5. Factores técnicos. . . . .	114
1.6. Factores ambientales. . . . .	115
1.7. Valores de LOE. . . . .	116
3.1. Prueba bloque ID 1. . . . .	126
3.2. Prueba bloque ID 2. . . . .	127
3.3. Prueba bloque ID 3. . . . .	128
3.4. Prueba bloque ID 4. . . . .	129

# Introducción

Este proyecto busca satisfacer la necesidad que presenta el departamento de geofísica de la Universidad de Concepción, quienes realizan estudios meteorológicos y desean hacer llegar dicha información a la comunidad regional de manera amigable aprovechando las capacidades que ofrecen los teléfonos inteligentes, a través de una aplicación para la plataforma móvil “Android”.

El documento está dividido en una serie de 11 capítulos en los cuales se describe todo lo realizado para llevar a cabo los objetivos planteados, considerando lenguajes aprendidos, herramientas utilizadas para el desarrollo e investigaciones relativas a los temas involucrados.

A continuación se presenta una breve descripción de los capítulos que conforman este proyecto de título:

- **Definición de la Institución:** En este capítulo se describe a la organización en general, y también el departamento en particular con el que se trabaja. Como punto clave de este capítulo se encuentra la definición de la problemática.
- **Definición del proyecto:** Se presentan los objetivos generales y específicos del proyecto, como también el ambiente de ingeniería de software donde se describe la metodología de desarrollo a utilizar y las notaciones implementadas en el documento.
- **Estado del arte:** Este capítulo sirve de introducción a las tecnologías móviles, dando a conocer los diferentes sistemas operativos con los que se puede encontrar el usuario en el comercio de los teléfonos inteligentes, las cuotas de mercado global que presentan las plataformas móviles. Luego, se especifica sobre el sistema operativo Android, tratando su historia, versiones disponibles del sistema operativo, y como es el desarrollo en dicha plataforma. También se introduce al lector en el modelo WRF y su funcionamiento. Para

finalizar, se presentan algunas aplicaciones que tienen un objetivo similar a la que se propone en este proyecto.

- **Especificación de requerimientos de software:** Se plantean los alcances que tiene el producto, especificando que hará y que no, también se presentan los objetivos generales y específicos que cumple el software, y por último la descripción de cada requerimiento necesario para el desarrollo de la aplicación.
- **Estudio de factibilidad:** Se presenta el estudio de la factibilidad técnica, operativa y económica. Finalizando con un análisis general para concluir el estudio.
- **Análisis:** En este apartado se plantea el diagrama de casos de uso, los actores involucrados y las conexiones entre ellos y los casos de uso.
- **Diseño:** Este capítulo tiene la finalidad de mostrar la arquitectura funcional del software, y también los aspectos relacionados con la interfaz que presenta la aplicación.
- **Pruebas:** En este capítulo se especifican las pruebas que se realizan al software así como también los encargados de su ejecución y el calendario de planificación de las mismas.
- **Plan de implantación y puesta en marcha:** Se definen los pasos posteriores al desarrollo del software, lo que se traduce en la publicación y distribución del producto.
- **Resumen de esfuerzo requerido:** Se especifica el total de horas trabajadas en el proyecto.
- **Conclusiones:** Se plantean las conclusiones del proyecto considerando los objetivos alcanzados y requerimientos cumplidos.
- **Bibliografía:** Se definen las referencias y materiales de apoyo utilizados para el proyecto.

- **Anexos:** En este apartado se presentan secciones que podrían ser de itneres para el lector.

# Capítulo 1

## Definición de la institución

### 1.1. Descripción de la institución

#### 1.1.1. Antecedentes generales de la Institución

- **Nombre:** Universidad de Concepción.
- **Dirección:** Calle Victor Lamas 1290, Concepción, Octava Región, Chile.
- **Rubro:** Educación.
- **Productos-servicios que ofrece:** Carreras de pregrado y postgrado.

#### 1.1.2. Entorno

- **Competencia Directa:** Instituciones de educación Superior (Universidades).

#### 1.1.3. Misión

La misión de la Universidad de Concepción es la formación integral y pluralista de personas con responsabilidad social, creatividad, sentido crítico, liderazgo y emprendimiento; la creación, adaptación y transmisión de conocimientos, y la creación y difusión cultural, con el propósito de contribuir efectivamente al desarrollo humano, económico y social sustentable de la región y el país. [1]

#### **1.1.4. Visión**

Universidad con creciente reconocimiento nacional e internacional por su calidad en la formación de personas y en investigación, desarrollo e innovación, con capacidad de anticipación y adaptación a los cambios del entorno, comprometida con la cultura y el desarrollo de la región y del país. [1]

#### **1.1.5. Organigrama**

A continuación se presenta el organigrama de la corporación (figura 1.1) a la que pertenece la casa de estudios con la que se realiza el trabajo.

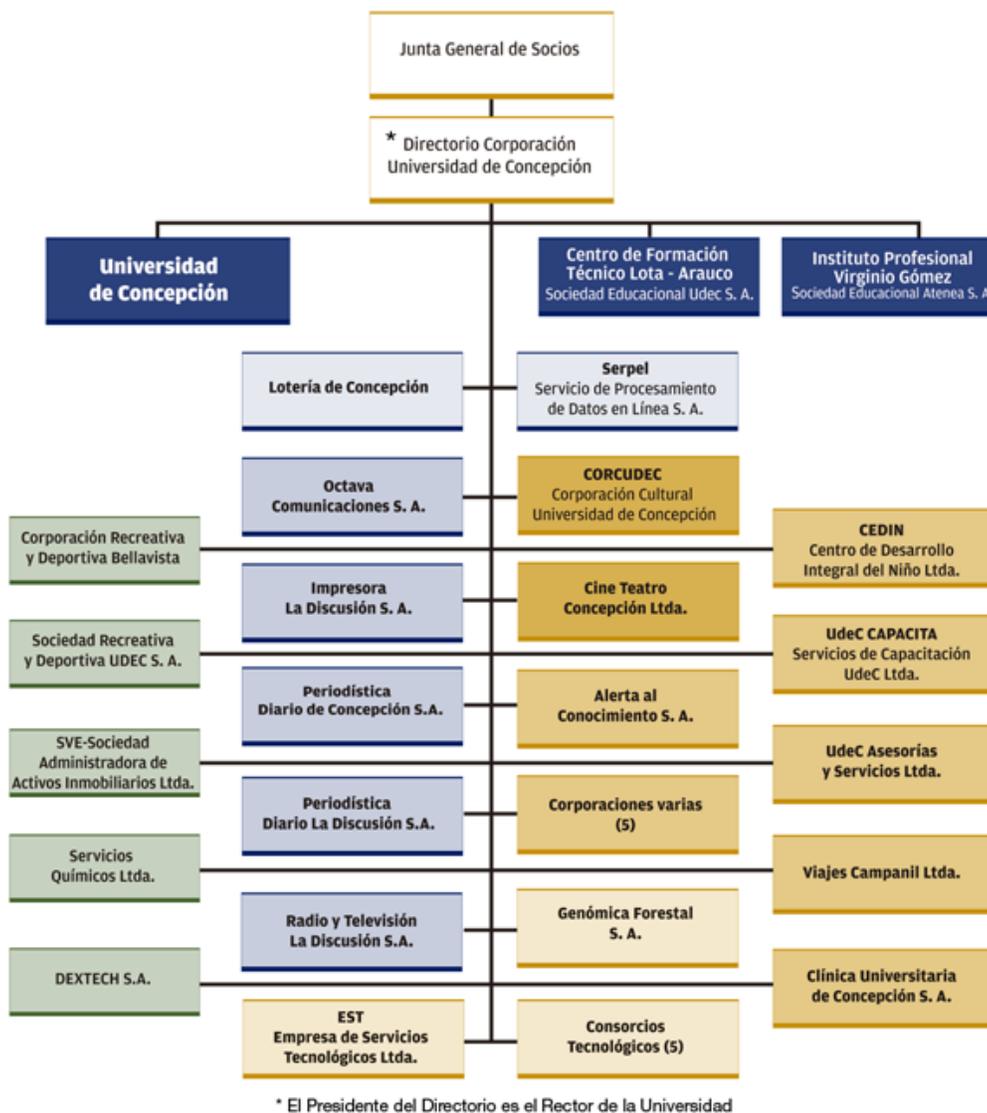


Figura 1.1: Organigrama Corporación UdeC.

Fuente: Página web. [2]

El siguiente diagrama 1.2 es el correspondiente al organigrama de la Universidad de Concepción.

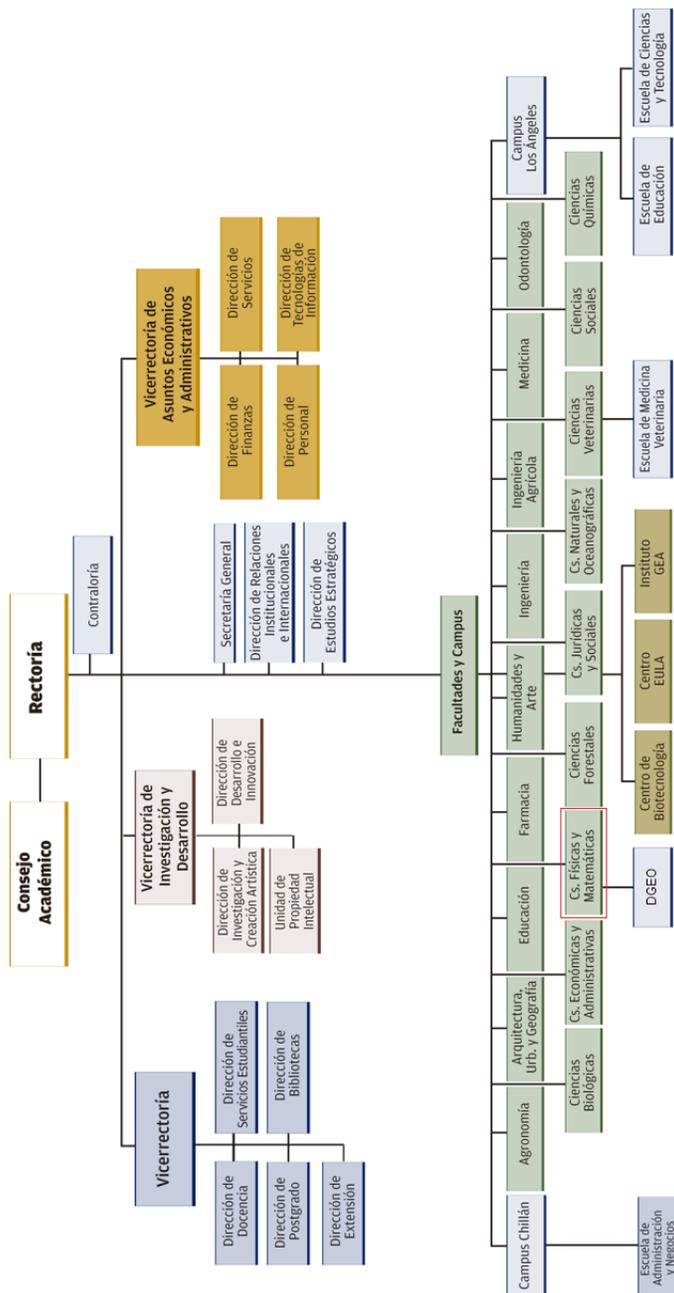


Figura 1.2: Organigrama Universidad de Concepción.

Fuente: Página web. [3]

El departamento de Geofísica (DGEO) que es con el que se trabaja, forma parte de la facultad de Ciencias Físicas y Matemáticas.

## 1.2. Descripción del área de estudio

El departamento de Geofísica (DGEO) siempre ha potenciado la investigación científica de la climatología y realizó la primera descripción de la región del Biobío en cuanto oceanografía física, en 1967, gracias al doctor en Meteorología Jean-Luc Devynck, que impulsó la investigación científica del departamento. El desafío del departamento de Geofísica de la Universidad de Concepción siempre ha sido contribuir de manera decisiva para dar respuesta a los impactos de los fenómenos naturales en los distintos sectores de la sociedad [4]. Dentro del departamento de geofísica se cuenta con la colaboración del señor Osvaldo Artal, quien en términos formales corresponde al profesor co-guía del proyecto.

## 1.3. Descripción de la problemática

### 1.3.1. Introducción

El departamento de Geofísica de la Universidad de Concepción, con sus estudios espera servir a la comunidad regional y a través de este proyecto crear lazos cooperativos con la Universidad del Bio-Bio.

En el departamento de Geofísica una de las herramientas que utilizan es el WRF (The Weather Research and Forecasting Model) que significa “Modelo de Investigación y pronósticos climáticos” el cual es un modelo de origen estadounidense y como su nombre lo dice su objetivo es el estudio del clima y los diversos factores asociados. Modelo que es implementado para las investigaciones y simulaciones por parte de los académicos e investigadores del departamento.

Como se puede ver a continuación, las figuras (1.3, 1.4, 1.5, 1.6) muestran algunos gráficos que se pueden obtener a través del modelo WRF, pero la información no es de fácil interpretación para quienes no son entendidos en la materia, idea de donde se origina la problemática.

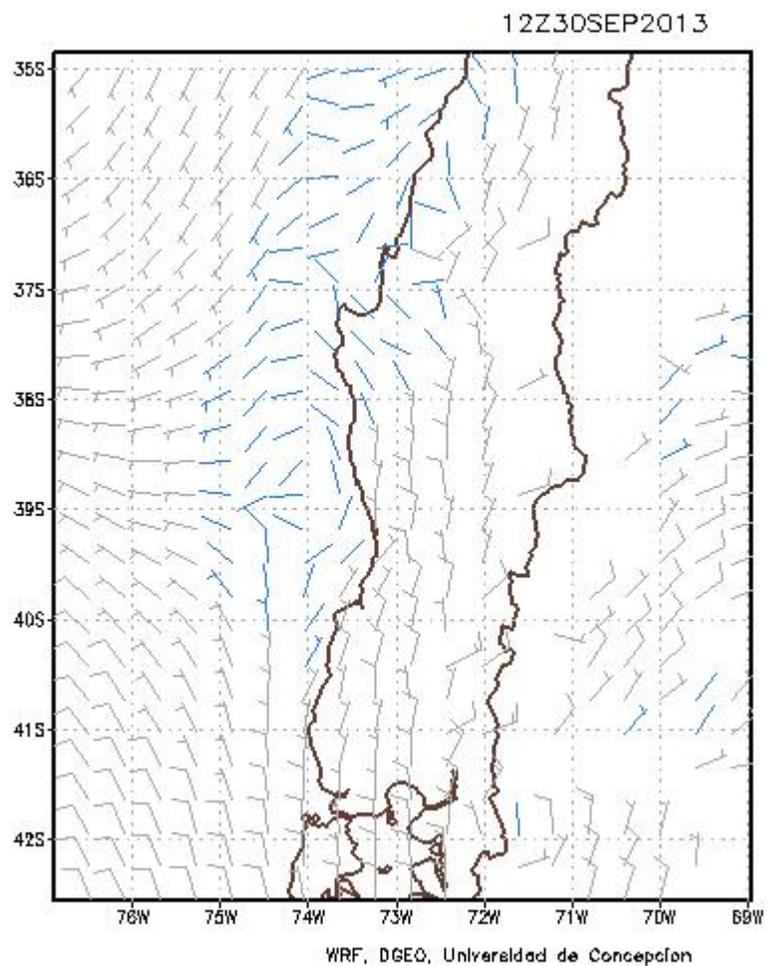


Figura 1.3: Variables de viento.

**Fuente:** Página web. [5]

La figura 1.3 presenta las variables de viento en un dominio específico e instante de tiempo determinado.

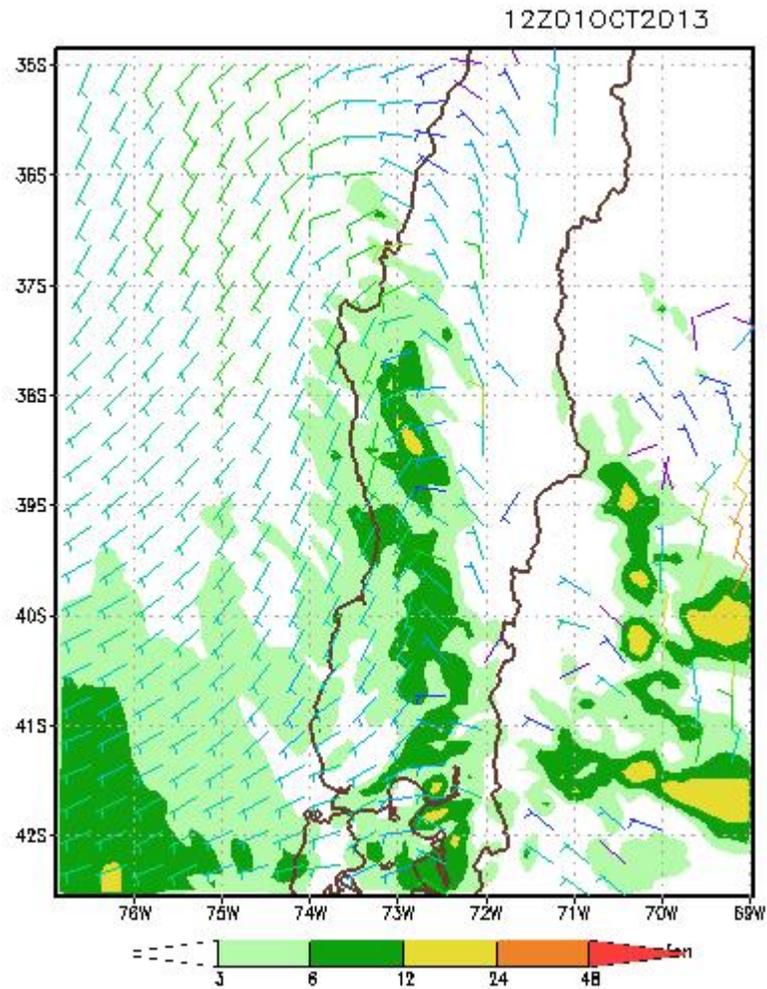


Figura 1.4: Precipitaciones.

**Fuente:** Página web. [5]

La figura 1.4 también presenta las variables de viento (representadas por las líneas rectas) además de los colores con los que se distingue las precipitaciones a partir de la escala presentada en la parte inferior del gráfico.

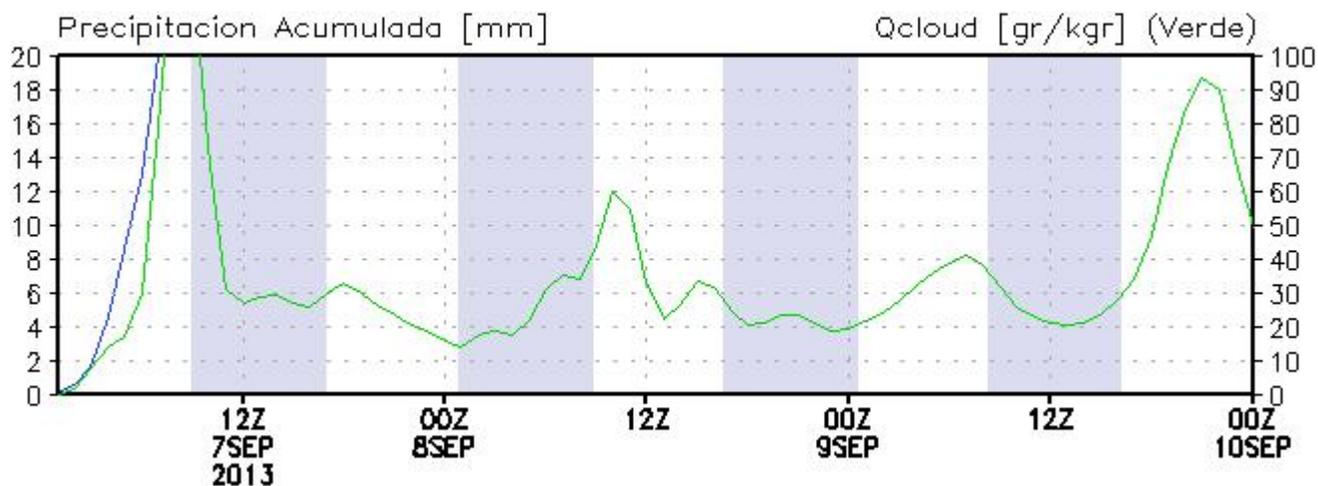


Figura 1.5: Temperatura y viento.

Fuente: Página web. [5]

La figura 1.5 presenta las precipitaciones acumuladas en milímetros y la variable Qcloud referente al cumulo de agua en la nube para el intervalo de tiempo dado.

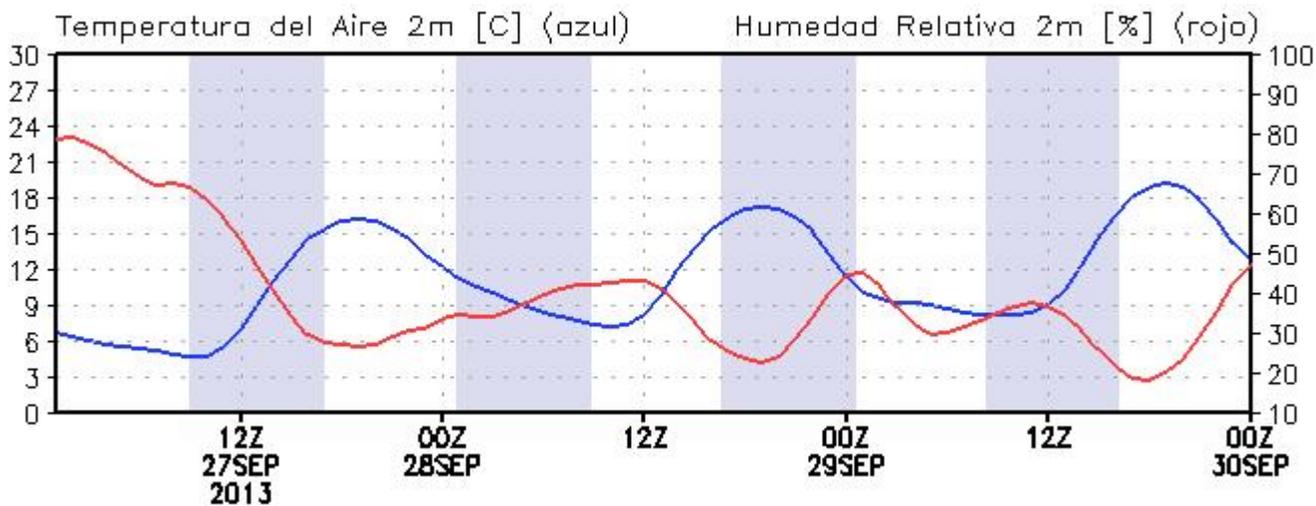


Figura 1.6: Temperatura y humedad.

Fuente: Página web. [5]

La figura 1.6 presenta en la gráfica de un intervalo de tiempo la relación entre la temperatura del aire y la humedad relativa.

### 1.3.2. La problemática

El modelo de predicción climática WRF, que ocupa el Departamento de Geofísica para sus estudios meteorológicos y oceanográficos, genera una inmensa cantidad de datos (varios Gigabyte) que se almacenan en unos archivos de extensión “.nc” referente a NetCDF (Network Common Data Form) que es un conjunto de librerías y formatos de datos independientes de la máquina de auto-descripción que apoyan la creación, el acceso y el intercambio de datos científicos. [6]

Al ejecutar el modelo, se descarga los datos provenientes de los satélites en órbita que cumplen con la función de recopilar la información, esta información es preprocesada para convertirla en archivos NetCDF. Una vez hecho esto el usuario del modelo puede realizar las tareas que desee, como por ejemplo, desplegar gráficas según las variables que estime para el caso (gráficas como las imágenes 1.3, 1.4, 1.5, 1.6 anteriormente presentadas).

Teniendo en cuenta estos antecedentes, es que el departamento ha decidido entregar esta información a la comunidad de la región, a través de una aplicación que sea para dispositivos con sistema operativo Android, la decisión de la plataforma escogida por el departamento, será respaldada por la investigación referente a los diferentes sistemas operativos móviles existentes, información que se presentará en el capítulo llamado “Estado de arte”.

### 1.3.3. Solución

La solución comprende una aplicación móvil para dispositivos con sistema operativo Android, la cual tomará los datos del modelo que sean más relevantes para el usuario acorde a los requerimientos establecidos por el departamento de Geofísica entre los cuales destacan: Temperatura, humedad, viento, etc.

El usuario puede consultar el pronóstico desde su dispositivo Android y recibir la información relativa al pronóstico solicitado de forma clara y precisa.

## Capítulo 2

# Estado del arte

### 2.1. Introducción

En este capítulo se presenta los estudios que se consideran relevantes para llevar a cabo el presente trabajo y lograr los objetivos propuestos. Dentro de estos puntos se encuentra el sistema operativo con el cual se trabaja (Android), para que el lector se familiarice con esta plataforma. También se incluye el modelo de investigación y previsión del tiempo (WRF), utilizado por el Departamento de Geofísica de la Universidad de Concepción, dando a conocer su funcionamiento y el manejo de datos. Se incluye la herramienta de software matemático, MATLAB, usada para la extracción de los datos del modelo WRF, a un formato legible y generar así la comunicación con la aplicación en Android. Finalmente se mencionan algunas aplicaciones similares disponibles en el mercado.

### 2.2. Dispositivos móviles inteligentes

#### 2.2.1. Plataformas disponibles

Los sistemas operativos móviles más usados en los últimos años son los siguientes:

- **Windows Phone:** Es el sistema operativo móvil desarrollado por Microsoft como sucesor de Windows Mobile. Windows Phone ofrece una nueva interfaz de usuario que integra varios de sus servicios propios como SkyDrive, Skype y Xbox Live en el sistema operativo. [7]

- **Android:** El desarrollo de este sistema operativo está liderado por Google Inc. Está basado en Linux, un núcleo de sistema operativo libre, gratuito y multiplataforma. Es uno de los sistemas operativos para dispositivos móviles más populares en la actualidad, debido a su gran flexibilidad de requerimientos de hardware, pudiéndose encontrar teléfonos a bajo costo pero con una gran gama de herramientas y funcionalidades. Existen muchas aplicaciones para ser descargadas en una tienda virtual “Google Play”. [9]
  
- **iOS:** Es el sistema operativo que da vida a dispositivos como el iPhone, el iPad, el iPod Touch o el Apple TV. Apple no permite que iOS esté presente en dispositivos de terceras compañías. La interfaz de usuario de este sistema operativo está basada en el concepto de manipulación directa, usando gestos multitáctiles. Posee una tienda virtual para descargar aplicaciones llamada “Apps Store”. [8]
  
- **BlackBerry OS:** Es el sistema operativo creado por la empresa Research In Motion (RIM). Su desarrollo, en primera instancia, fue enfocado para el ambiente empresarial o de oficina, pero se ha tenido que expandir a otros ámbitos debido a la fuerte competencia de iPhone y Android, llegando a tener su propia tienda virtual de aplicaciones llamada “BlackBerry App Worl”. [10]
  
- **Otros:** En menor medida o como proyectos futuros se encuentran los sistemas de Firefox, Ubuntu Mobile, Symbian (descontinuado).

En la Tabla 2.1 se puede apreciar un cuadro comparativo entre las cuatro plataformas presentadas, en la cual se puede observar que en la plataforma iOS es mucho más difícil el desarrollo, ya que Apple cuenta con un proceso de “postulación” para poder publicar las aplicaciones en su App Store, a las cuales una vez aprobadas se les entrega un certificado, cabe destacar además que no se permite que se instalen aplicaciones sin certificados de aprobación por parte de Apple, por lo cual si no se obtiene dicho certificado, no podrá ser comercializada la aplicación hasta que se resuelvan los problemas encontrados.

Por otro lado, las demás plataformas ofrecen una gran facilidad para la publicación de aplicaciones, ya que no cuentan con el filtro de publicación como en iOS. Se puede apreciar también que el grado de personalización en Android es mayor que en todas las demás plataformas, al igual que la gestión de los datos en tiempo real.

 iPhone iOS 7.0.4	 Android 4.2 Jelly Bean	 Blackberry OS 10	 Windows Phone 8
<p><b>General</b></p> <p><b>Positivo:</b> Apple Controla el Hardware y Software lo que asegura un mejor trabajo entre estos dos elementos.</p> <p><b>Negativo:</b> iOS proporciona una experiencia del uso del smartphone controlada por lo que si se desea remover alguna restricción, es posible que se le bloquee la cuenta al usuario.</p>	<p><b>General</b></p> <p><b>Positivo:</b> Android posee la mayor parte del mercado y la experiencia del usuario está mejorando rapidamente.</p> <p><b>Negativo:</b> Las actualizaciones son un problema ya que la mayoría de los aparatos móviles Android se encuentran atrasados en varias versiones.</p>	<p><b>General</b></p> <p><b>Positivo:</b> Proporciona seguridad en el envío de mensajes ya que posee un alto nivel de encriptación, además, los dispositivos con este sistema son los primeros en su tipo en poseer pantalla táctil.</p> <p><b>Negativo:</b> es un sistema operativo muy nuevo, lo que queda demostrado en las pocas aplicaciones existentes.</p>	<p><b>General</b></p> <p><b>Positivo:</b> Muy similar a utilizar el escritorio de Windows 8 en un computador con ese sistema operativo.</p> <p><b>Negativo:</b> Pequeña cuota de mercado lo que presenta un bajo soporte en cuanto a las aplicaciones se trata.</p>
<p><b>Hardware</b></p> <p><b>Positivo:</b> Buen rendimiento y estabilidad.</p> <p><b>Negativo:</b> Diseño casi no ha variado en años; la batería no es reemplazable y la memoria no es expandible, limitando al usuario a escoger dentro de las versiones que se presenten.</p>	<p><b>Hardware</b></p> <p><b>Positivo:</b> Gran cantidad de modelos para escoger de diferentes marcas; la batería es reemplazable y la memoria es expandible en la mayoría de los dispositivos.</p>	<p><b>Hardware</b></p> <p><b>Positivo:</b> La memoria es expandible.</p> <p><b>Negativo:</b> ya no cuenta con la misma batería de larga duración que tenía en el pasado y hay muy poca variedad de dispositivos en el mercado.</p>	<p><b>Hardware</b></p> <p><b>Positivo:</b> memoria expandible. Microsoft tiene requisitos estrictos en cuanto a la calidad de hardware.</p> <p><b>Negativo:</b> En el mercado existe poca variedad de dispositivos.</p>
<p><b>Personalización</b></p> <p><b>Negativo:</b> No es posible instalar aplicaciones que no sean de la tienda de aplicaciones de Apple, ni tampoco versiones modificadas del sistema operativo.</p>	<p><b>Personalización</b></p> <p><b>Positivo:</b> Es altamente modificable, se pueden instalar versiones no oficiales del SO y aplicaciones que no pertenezcan a Google Play.</p> <p><b>Negativo:</b> Existen demasiadas versiones de Android lo que genera confusión en los usuarios.</p>	<p><b>Personalización</b></p> <p><b>Positivo:</b> Se pueden modificar ciertos elementos, pero no de la misma forma que en Android.</p>	<p><b>Personalización</b></p> <p><b>Positivo:</b> Pantalla de inicio altamente configurable.</p>

Tabla 2.1: Comparación entre sistemas operativos móviles.

Fuente: Elaboración propia.

### 2.2.2. Justificación plataforma escogida

Según la información recopilada y las comparaciones realizadas, se considerarán los siguientes criterios para desarrollar la aplicación en el sistema operativo Android:

- Si bien fue Apple el que supo suplir la necesidad de las tiendas de aplicaciones con su App Store, seguido luego por Google play (tienda de Android), fue rápidamente alcanzado y superado por este último en cuanto a la cantidad de aplicaciones existentes, esto debido a que para instalar una aplicación en un iPhone o iPads de forma legal tiene que ser a través de su App Store ya que al ser un sistema operativo cerrado no permite que sus equipos tengan aplicaciones que no hayan sido previamente validadas para pertenecer a la App Store. En cambio en Android se puede instalar lo que se desee, es decir, no hay porqué pasar por su plataforma de distribución.
- Otro punto a favor para Android es el hecho de su tendencia de crecimiento y el mercado global que este ocupa, llegando al 69,1% el año 2012 según el IDC, porcentaje que ha incrementado en el transcurso del año 2013, lo que establece una tendencia muy fuerte por parte del sistema operativo móvil de Google a nivel mundial.

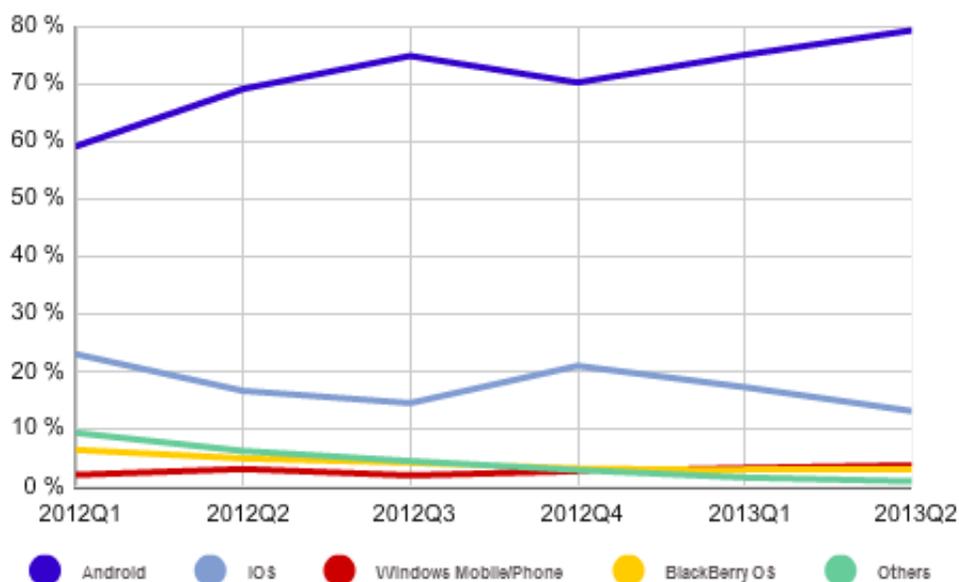


Figura 2.1: Cuota de Mercado global de plataformas de smartphone.

**Fuente:** Página web [14].

- Otra razón por la que no se prefiere otro sistema operativo para desarrollar, es que tanto Windows phone 8, Blackberry OS 10 y también iOS son sistemas operativos de software propietario (no es de acceso gratuito) lo que obliga al desarrollador a adquirir algún tipo de herramienta pagada ( Mac OS X, IDE pagado, etc.-). Al poder desarrollar para la plataforma Android sin tener que invertir en elementos, facilitando el trabajo, y así utilizar sin ningún problema los equipo que se poseen.

## 2.3. Sistema operativo Android

### 2.3.1. Introducción a Android

Android es un sistema operativo para dispositivos móviles. Está basado en GNU/Linux e inicialmente fue desarrollado por Android Inc. que después fue comprada por Google en el año 2005 y poco después se unió “Open Handset Alliance”, un consorcio de 48 compañías de hardware, software y telecomunicaciones.

Android permite controlar dispositivos por medio de librerías desarrolladas o adaptadas por Google mediante el lenguaje de programación Java.

Es una plataforma de código abierto. Esto quiere decir que cualquier desarrollador puede crear aplicaciones que serán de acceso libre para el resto de las personas, siempre y cuando se siga la filosofía del código abierto. [9]

### 2.3.2. Versiones de Android

Android posee numerosas versiones y actualizaciones que se inició con el primer lanzamiento “Android beta”, estas actualizaciones normalmente corrigen fallos de programa y agregan nuevas funcionalidades. Es muy importante observar que cada versión que ha sido lanzada siempre ha sido compatible con las otras anteriores y en el caso de modificar alguna funcionalidad no se elimina, se etiquetan como obsoletas pero se pueden seguir utilizando.

Desde abril de 2009, las versiones de Android han sido desarrolladas bajo un nombre y lanzadas en orden alfabético.

En la siguiente tabla (2.2), se puede observar que tiene una columna llamada “N° de API”, el cual es un valor entero que representa un conjunto de funciones y procedimientos, que sirven para comunicarse con las versiones anteriormente lanzadas. Este identificador, que distingue de forma exclusiva la versión de la plataforma, permite establecer cuál será la versión mínima y máxima que soporte una aplicación Android. [9]

Versión	Nº de API	Lanzamiento
Android 4.4	19	Octubre 2013
Android 4.3	18	Julio 2013
Android 4.2, 4.2.2	17	Noviembre 2012 Enero 2013 (4.2.2)
Android 4.1, 4.1.1	16	Julio 2012
Android 4.0.3, 4.0.4	15	Diciembre 2011
Android 4.0, 4.0.1, 4.0.2	14	Octubre 2011
Android 3.2	13	Junio 2011
Android 3.1.x	12	Mayo 2011
Android 3.0.x	11	Febrero 2011
Android 2.3.4	10	Febrero 2011
Android 2.3.3		
Android 2.3.2	9	Noviembre 2010
Android 2.3.1		
Android 2.3		
Android 2.2.x	8	Junio 2010
Android 2.1.x	7	Enero 2010
Android 2.0.1	6	Diciembre 2009
Android 2.0	5	Noviembre 2009
Android 1.6	4	Septiembre 2009
Android 1.5	3	Mayo 2009
Android 1.1	2	Febrero 2009
Android 1.0	1	Octubre 2008

Tabla 2.2: Tabla con versiones y Nº de API.

Fuente: Elaboración propia.

En la figura que se presenta a continuación se puede apreciar cuales son las versiones más utilizadas por dispositivos Android.

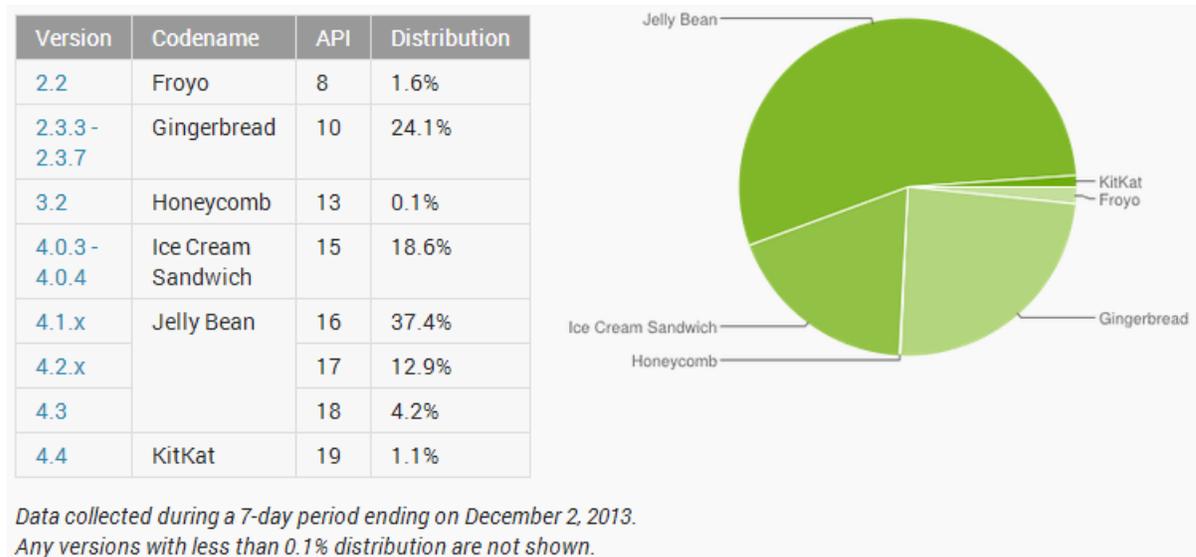


Figura 2.2: Porcentaje de utilización de las versiones existentes en Android.

**Fuente:** Página web.

### 2.3.3. Componentes fundamentales

En este apartado se describen los pasos básicos en conjunto con las herramientas necesarias para comenzar a programar aplicaciones para la plataforma Android. El entorno de desarrollo que se escoge es Eclipse, ya que los alumnos tesisistas están familiarizados con este IDE.

#### Entorno de desarrollo

Teniendo instalado el IDE Eclipse, no es suficiente para comenzar a desarrollar una aplicación para la plataforma Android, para esto se necesita el SDK, ya que este proporciona las bibliotecas API y las herramientas de desarrollo necesarias para crear, probar y depurar aplicaciones para Android. Para su instalación solo basta con ejecutar el instalador estándar de Windows. [11]

Además del SDK de Android, que contiene las herramientas básicas para desarrollar aplicaciones en Android, también se necesita descargar las llamadas Platform Tools 2.3, que contiene herramientas específicas de la última versión de la plataforma, y una o varias plataformas (SDK Platforms) de Android, que no son más que las librerías necesarias para desarrollar sobre cada

una de las versiones concretas de Android. Es decir, si se desea desarrollar por ejemplo para Android 2.2 se debe que descargar su plataforma correspondiente.

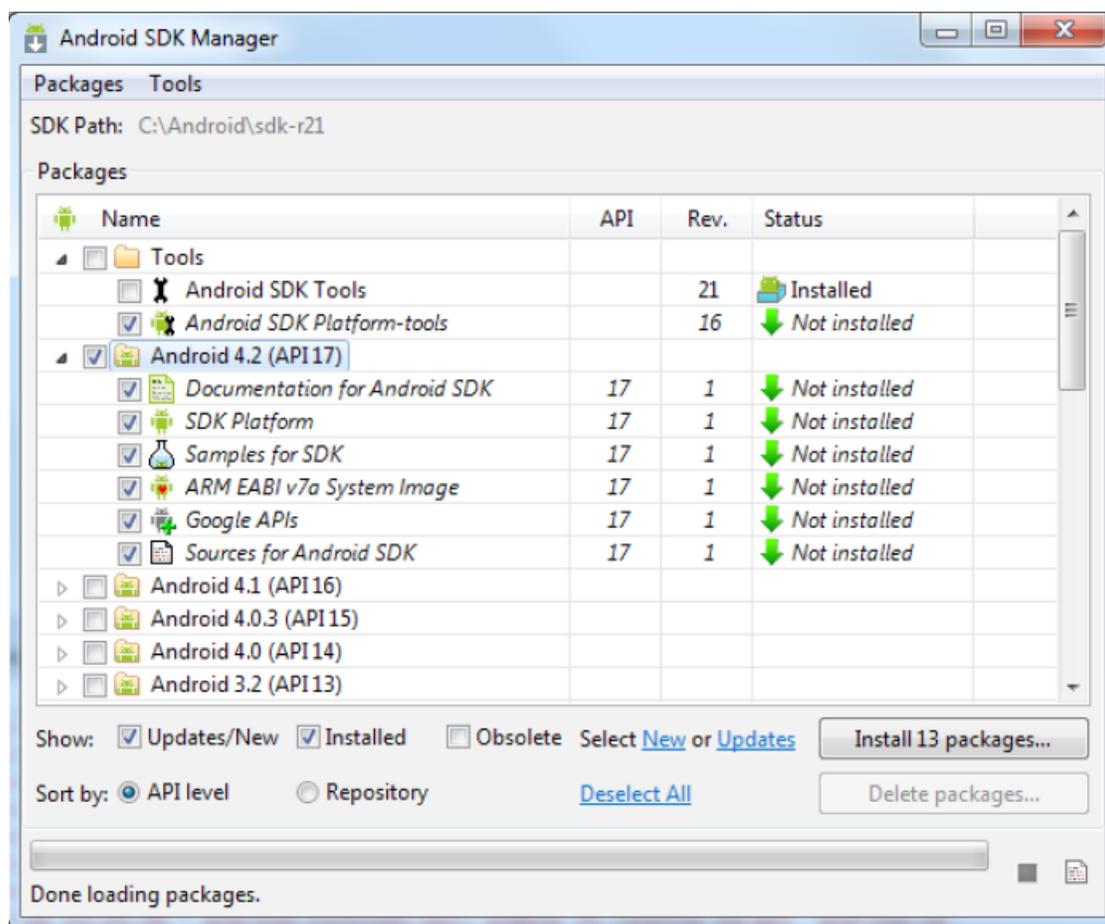


Figura 2.3: Menu del SDK Manager para escoger la plataforma Android a utilizar.

**Fuente:** Elaboración propia.

Así también Google pone a disposición de los desarrolladores un plugin para Eclipse llamado Android Development Tools (ADT) que facilita en gran medida el desarrollo de aplicaciones para la plataforma, ampliando las capacidades de Eclipse.

Finalmente si así se desea, a la hora de probar y depurar aplicaciones Android no es necesario hacer esto sobre un dispositivo físico, sino que se puede configurar un emulador o dispositivo virtual (Android Virtual Device o AVD) para poder realizar fácilmente estas tareas. Para ello, se accede al AVD Manager (menú Window / AVD Manager), y en la sección Virtual Devices se

añaden tantos AVD como se necesiten (figura 2.4 ).

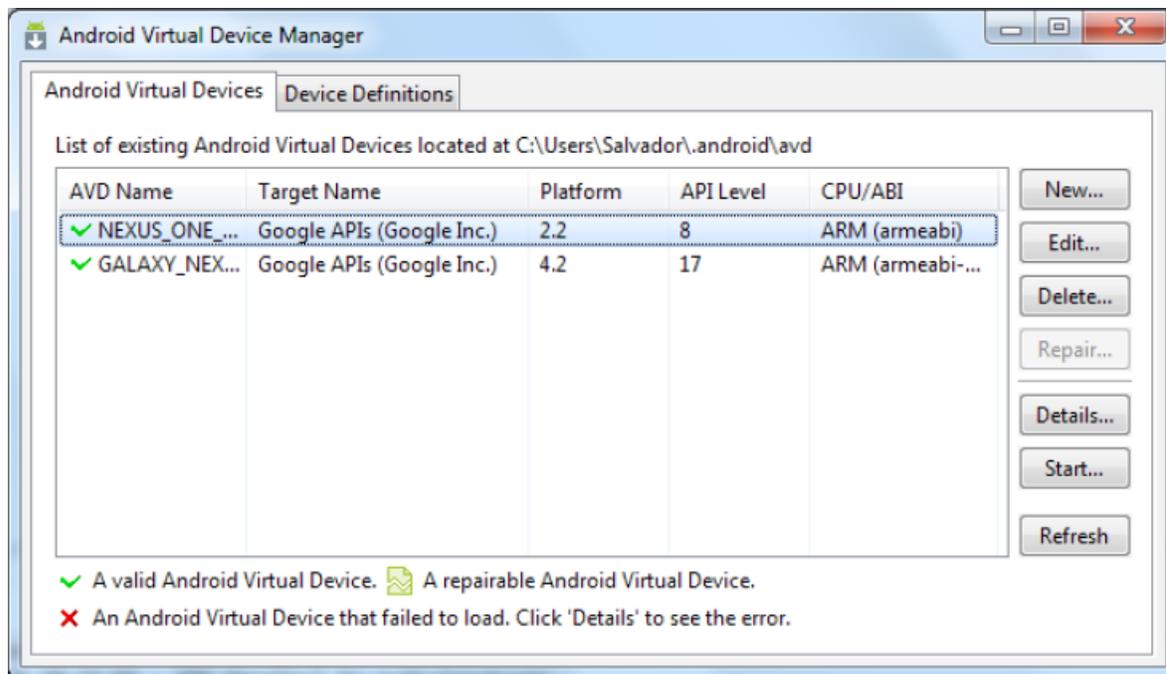


Figura 2.4: Android Virtual Devices.

**Fuente:** Elaboración propia.

Se puede apreciar la lista de todos los AVD disponibles, con la opción de agregar nuevos emuladores, por medio del botón “New...”. A modo de ejemplo se puede observar que se ha creado un emulador con API level 8, es decir emulará las características de un dispositivo Android 2.2 y el emulador con API level 17 emulará las de un dispositivo Android 4.2.

Al momento de crear un emulador se pueden configurar varias de sus características como por ejemplo las que siguen a continuación:

- **AVD Name:** Corresponde al nombre del emulador. Se recomienda utilizar un nombre que haga referencia a la versión de Android que se emula, en la imagen 2.5 se puede ver que en el presente proyecto lo nombran Android4.3. [12]
- **Device:** Permite elegir el dispositivo que se utilizara en cuanto a resolución. [12]
- **Target:** Indica la versión de API de Android [12].
- **Front Camera:** Si se requiere emular la cámara delantera se marca esta opción. [12]

- 
- **Back Camera:** En caso de utilizar la cámara trasera se marca esta opción para emularla. [12]
  - **Memory Options:** Aquí se indica el tamaño de la memoria RAM y de VM Heap, este ultimo corresponde a la capacidad de almacenamiento y sirve cuando se utilizan aplicaciones que requieren mayor memoria. [12]
  - **Internal Storage:** Es el tamaño total de la memoria interna del dispositivo. [12]
  - **SD Card:** Es la memoria total que se asigna a la tarjeta SD del dispositivo. [12]
  - **Emulation Options:** La opción “Snapshot” permite arrancar el emulador más rápido para futuras ejecuciones y la opción “Use Host GPU” significa que en lugar de utilizar la CPU para procesar los gráficos, utiliza la tarjeta gráfica del ordenador. [12]

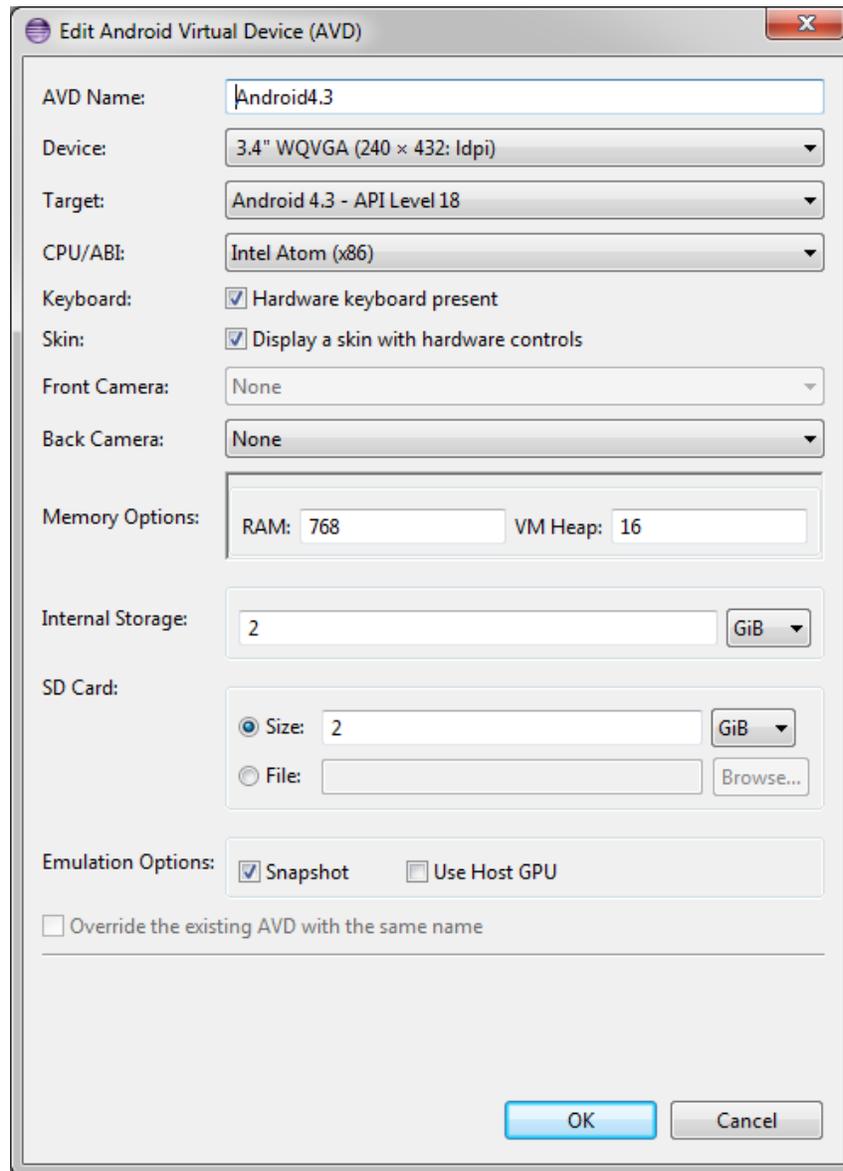


Figura 2.5: Opciones para editar el emulador en Android.

**Fuente:** Elaboración propia.

## Estructura de un proyecto en Android

Para comprender cómo se construye una aplicación Android se describirá como se crea un nuevo proyecto Android en Eclipse, revisando también la estructura general del proyecto creado por defecto.

Para crear un nuevo proyecto se abre Eclipse y se siguen los siguientes pasos:

1. En la barra de tareas seleccionar New
2. En la ventana que aparece seleccionar Proyecto y luego escoger la carpeta Android y elegir Android Application Project, y hacer clic en Siguiente.
3. Rellenar el formulario que aparece:
  - **Application name:** Es el nombre que se le dará a la aplicación
  - **Project Name:** Es el nombre del directorio del proyecto y el nombre visible en Eclipse.
  - **Package Name:** Es el paquete java que usa en las clases java.
  - **Minimum Required SDK:** Es la versión mínima del SDK que aceptará nuestra aplicación al ser instalada en un dispositivo.
  - **Target SDK:** Es la versión del SDK para la que se desarrollara.
  - **Compile With:** Corresponde a la versión del SDK con la que se compilará el proyecto.

**New Android Application**

Enter an application name (shown in launcher)

Application Name:

Project Name:

Package Name:

Minimum Required SDK: API 18: Android 2.2 (Froyo)

Target SDK: API 18: Android 4.3

Compile With: API 18: Android 4.3

Theme: Holo Light with Dark Action Bar

? < Back Next > Finish Cancel

Figura 2.6: Formulario 1.

**Fuente:** Elaboración propia.

- Al pulsar el botón Next, se accede al segundo paso, donde se debe indicar si durante la creación del nuevo proyecto se desea crear un icono para la aplicación “Create custom launcher icon” y si se quiere crear una actividad inicial “Create activity”. También se puede indicar si el proyecto será del tipo Librería “Mark this Project as a library”.

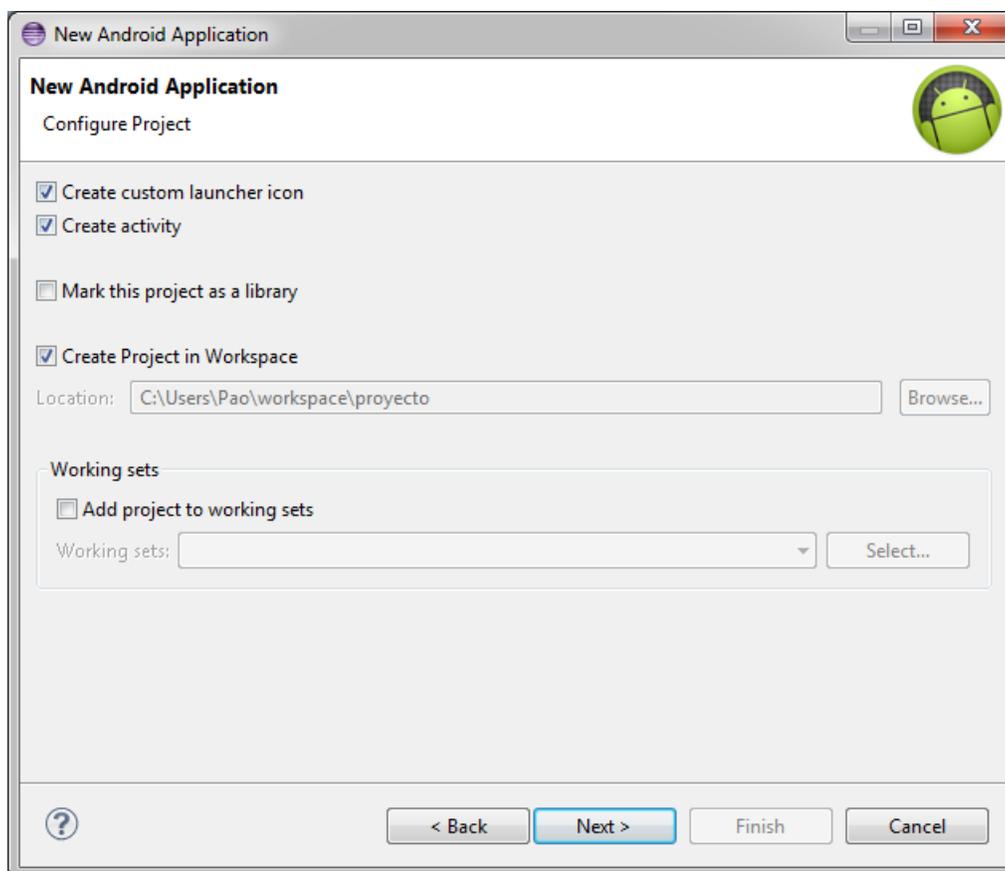


Figura 2.7: Formulario 2.

**Fuente:** Elaboración propia.

5. En la siguiente pantalla del asistente se puede configurar el icono que tendrá la aplicación en el dispositivo móvil. Se selecciona la imagen, texto o dibujo predefinido que aparecerá en el icono, el margen, la forma y los colores aplicados.

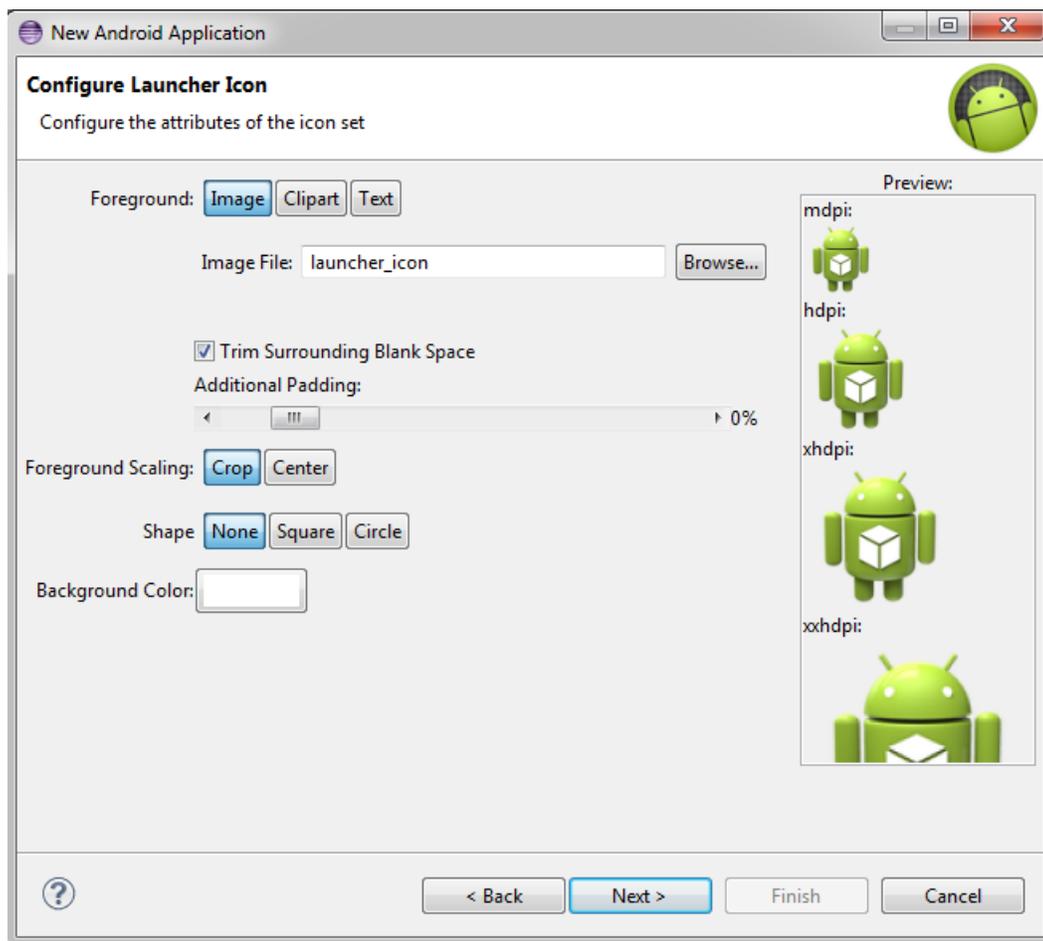


Figura 2.8: Formulario 3.

**Fuente:** Elaboración propia.

6. Luego de apretar “Next”, en la siguiente pantalla se elige el tipo de actividad principal de la aplicación. En este contexto se entiende que una actividad es una “ventana” o “pantalla” de la aplicación.

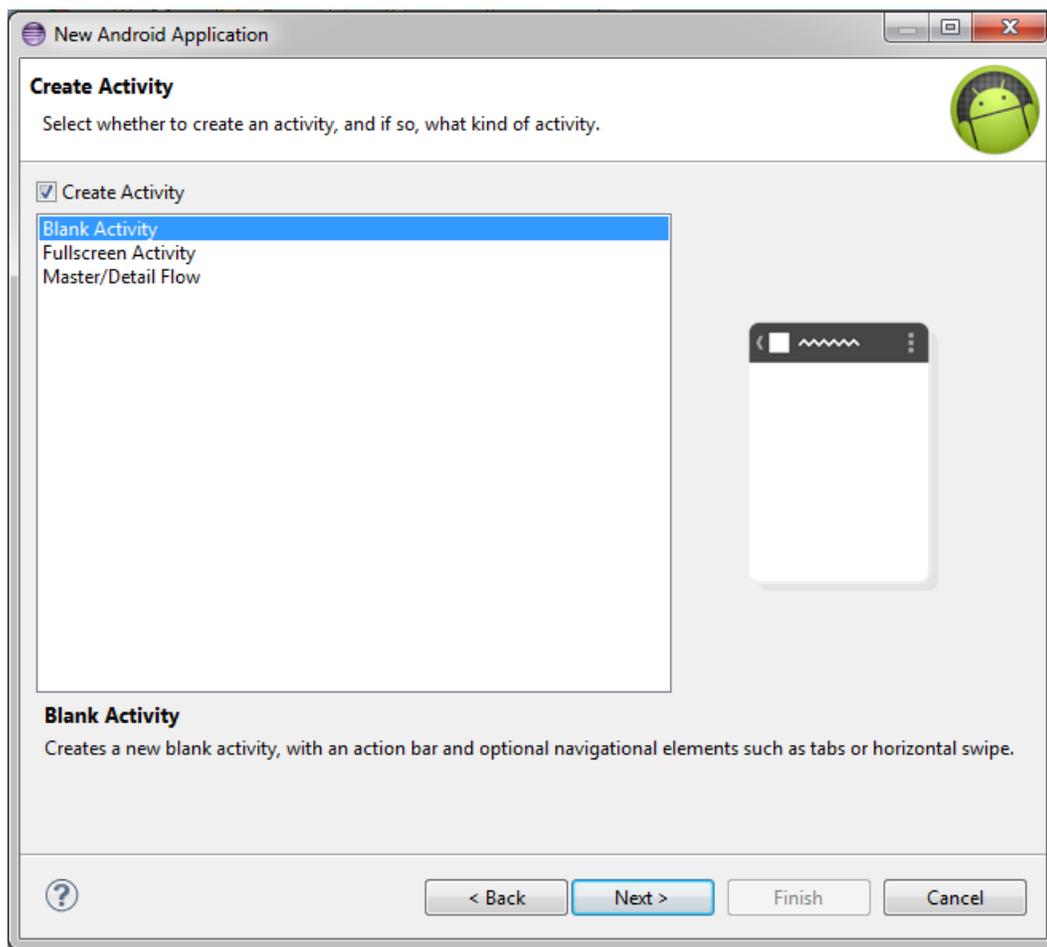
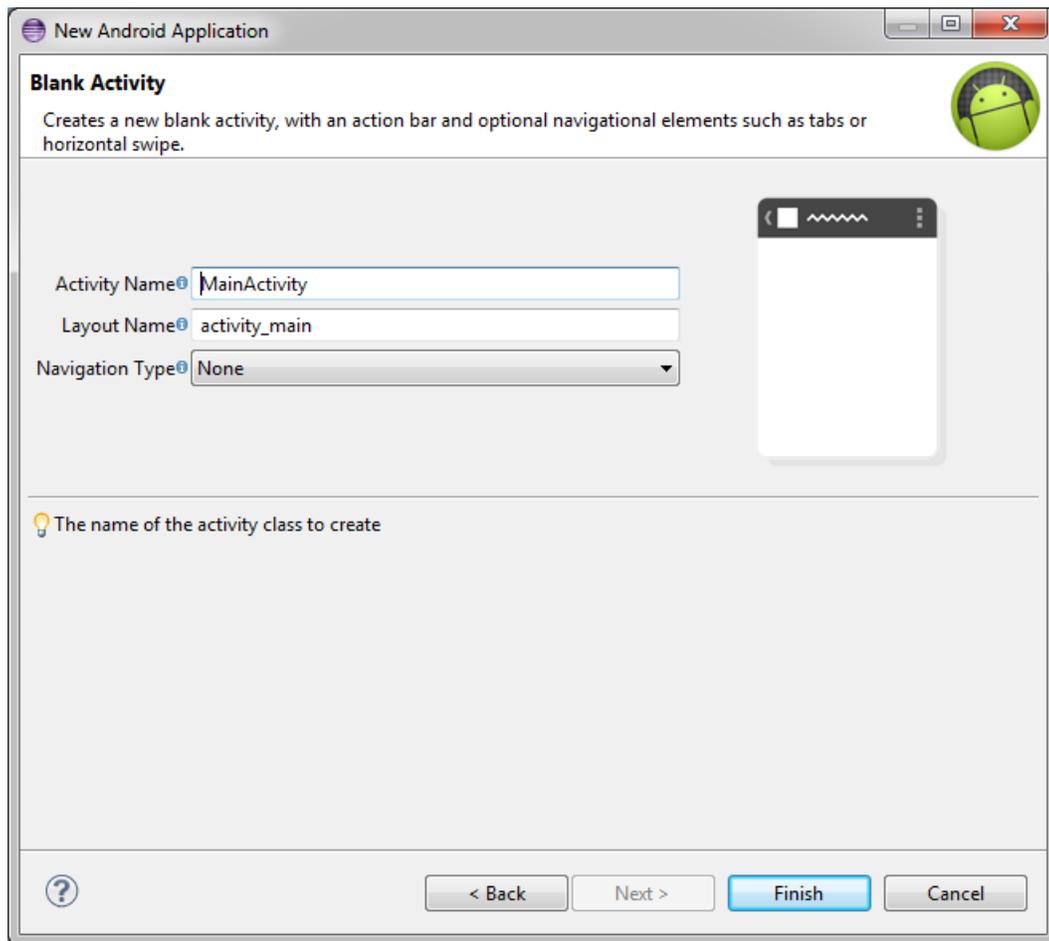


Figura 2.9: Formulario 4.

**Fuente:** Elaboración propia.

7. Finalmente, en el último paso se indican los datos de esta actividad principal que se acaba de elegir, se escribe el nombre de su clase java asociada y el nombre de su layout xml (interfaz gráfica de la actividad).



**New Android Application**

**Blank Activity**  
Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Activity Name: MainActivity

Layout Name: activity\_main

Navigation Type: None

The name of the activity class to create

< Back   Next >   Finish   Cancel

Figura 2.10: Formulario 5.

**Fuente:** Elaboración propia.

Al haber apretado el botón “Finish”, Eclipse creará automáticamente toda la estructura del proyecto y los elementos indispensables que debe contener. En la siguiente imagen 2.11 se puede ver los elementos creados inicialmente para un nuevo proyecto Android.

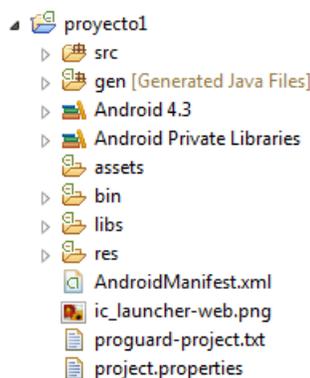


Figura 2.11: Estructura del proyecto en Android.

**Fuente:** Elaboración propia.

## Componentes de una aplicación Android

En este nuevo apartado se darán a conocer los componentes principales que pueden formar parte de una aplicación Android.

- **Activity:** Las actividades representan el componente principal de la interfaz gráfica de una aplicación Android. Se puede pensar en una actividad como el elemento análogo a una ventana o pantalla en cualquier otro lenguaje visual.
- **View:** Las vistas son los componentes básicos con los que se construye la interfaz gráfica de la aplicación. De inicio, Android pone a disposición una gran cantidad de controles básicos, como cuadros de texto, botones, listas desplegables o imágenes, aunque también existe la posibilidad de extender la funcionalidad de estos controles básicos.
- **Service:** Los servicios son componentes sin interfaz gráfica que se ejecutan en segundo plano. En concepto, son similares a los servicios presentes en cualquier otro sistema operativo. Los servicios pueden realizar cualquier tipo de acciones, por ejemplo actualizar datos, lanzar notificaciones, o incluso mostrar elementos visuales si se necesita en algún momento la interacción con del usuario.

- **Content Provider:** Un content provider es el mecanismo que se ha definido en Android para compartir datos entre aplicaciones. Mediante estos componentes es posible compartir determinados datos de la aplicación sin mostrar detalles sobre su almacenamiento interno, su estructura, o su implementación. De la misma forma, la aplicación podrá acceder a los datos de otra a través de los content provider que se hayan definido.
  
- **Broadcast Receiver:** Un broadcast receiver es un componente destinado a detectar y reaccionar ante determinados mensajes o eventos globales generados por el sistema (por ejemplo: “Batería baja”, “SMS recibido”, “Tarjeta SD insertada”) o por otras aplicaciones (cualquier aplicación puede generar mensajes o intents, como se conocen en Android).
  
- **Widget:** Los widgets son elementos visuales, normalmente interactivos, que pueden mostrarse en la pantalla principal del dispositivo Android y recibir actualizaciones periódicas. Permiten mostrar información de la aplicación al usuario directamente sobre la pantalla principal.
  
- **Intent:** Se pueden entender por intent como los mensajes o peticiones que son enviados entre los distintos componentes de una aplicación o entre distintas aplicaciones. Mediante un intent se puede mostrar una actividad desde cualquier otra, iniciar un servicio, enviar un mensaje broadcast, iniciar otra aplicación, etc.

## 2.4. WRF (Weather Research and Forecasting Model)

### 2.4.1. Introducción

La modelación atmosférica predice el estado del tiempo resolviendo mediante métodos numéricos las ecuaciones matemáticas para la física y la dinámica de la atmósfera a partir de condiciones previas.

Los modelos de predicción pueden ser globales, regionales (o de mesoescala <sup>1</sup>) según las escalas espaciales a las que se aplican.

---

<sup>1</sup>Mesoescala se encarga de estudiar a los fenómenos cuyas dimensiones espaciales van desde los kms a miles de kms y poseen ciclos de vida que abarcan los minutos hasta uno o dos días, ejemplo pueden ser los tornados.

El pronóstico del tiempo se realiza actualmente con la ayuda de uno o múltiples modelos numéricos, el modelo de Investigación Meteorológica y Predicción (WRF) es uno de ellos y se encuentra disponible de manera gratuita para la comunidad científica internacional.

El desarrollo de WRF ha sido un esfuerzo de varias agencias para construir un modelo de pronóstico de mesoescala de última generación y un sistema de asimilación de datos para avanzar en la comprensión y la predicción del tiempo.

Su creación ha sido una colaboración entre el National Center for Atmospheric Research's (NCAR) y la división de Mesoscale and Microscale Meteorology (MMM) , el National Oceanic and Atmospheric Administration's (NOAA) y los Centers for Environmental Prediction (NCEP) y el laboratorio Earth System Research Laboratory (ESRL), el departamento de defensa Air Force Weather Agency (AFWA) y el Naval Research Laboratory (NRL), el Center for Analysis and Prediction of Storms (CAPS) de la Universidad de Oklahoma, y la Federal Aviation Administration (FAA).

WRF se encuentra disponible de forma gratuita para la comunidad científica global, facilitando la investigación, las operaciones, y la enseñanza. Es usado en una amplia gama de aplicaciones. Estas aplicaciones incluyen el desarrollo de las simulaciones climáticas regionales.[19]

A continuación se presenta el diagrama WPS (WRF Preprocessing System) que prepara los archivos de entrada para la simulación con datos provenientes del WRF. También se da una pequeña reseña de los otros programas que están dentro del WPS.

#### 2.4.2. Funciones del WPS (WRF Preprocessing System)

El WPS prepara los archivos de entrada para las simulaciones con datos reales del WRF:

- Define los dominios de las simulaciones.
- Calcula la latitud, longitud en cada punto de la malla.
- Interpola<sup>2</sup> los datos independientes del tiempo en los puntos de la malla (e.j., la altura del terreno y el tipo de suelo).
- Interpola los campos meteorológicos dependientes del tiempo, provenientes de otros modelos.[19]

---

<sup>2</sup>Se denomina interpolación a la obtención de nuevos puntos partiendo del conocimiento de un conjunto de puntos, por ejemplo al tener dos puntos en un mapa cartesiano, se obtiene un tercer punto a través de una fórmula.

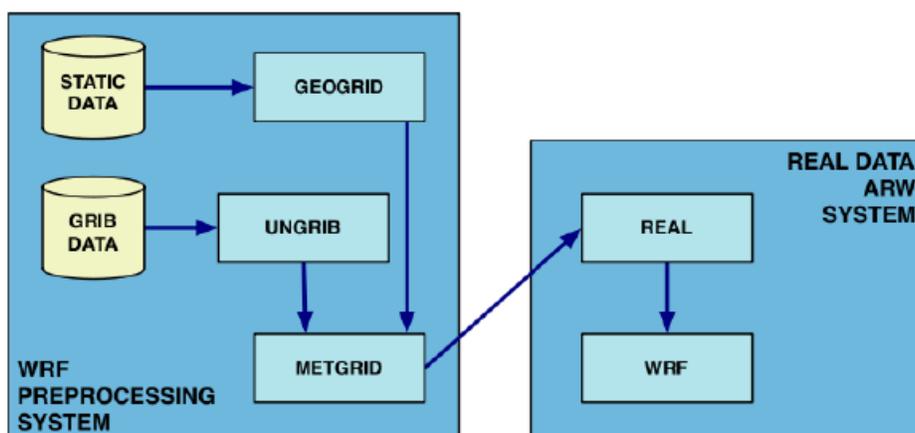


Figura 2.12: Esquema que muestra el flujo de datos y componentes del programa WPS y como los datos llegan finalmente al WRF. Las letras en los rectángulos indican los nombres de los programas que ayudan en el proceso.

**Fuente:** A Description of the Advanced Research WRF Version 3 [19]

### Programa Geogrid

El propósito del programa Geogrid es definir el tamaño y la localización de los dominios de simulación del modelo, e interpolar varios conjuntos de datos terrestres a las redes de modelo. El dominio de la simulación se define utilizando la información especificada por el usuario en el “Geogrid” del registro de la lista de nombres de archivos WPS, `namelist.wps`. De manera predeterminada, y además de la latitud y las longitudes para cada punto de la cuadrícula, Geogrid interpolará categorías de suelo, la altura del terreno, la temperatura media anual del suelo, vegetación mensual, porcentaje de radiación máximo que refleja la nieve. [19]

### Programa Ungrib

El programa Ungrib lee archivos provenientes del programa Geogrid, conocidos como los archivos GRIB, y escribe los datos en un formato más sencillo, llamado el “formato intermedio”. Los archivos GRIB contienen campos meteorológicos (variables en el tiempo) y suelen ser regionales o globales.[19]

### Programa Metgrid

El programa Metgrid interpola horizontalmente los datos meteorológicos del formato intermedio realizados por el programa Ungrib, sobre los dominios de simulación definidos por el

programa Geogrid. Así los datos de salida del programa Metgrid interpolados pueden ser leídos por el programa real.exe y entregados al WRF. Estos datos de salida del programa real.exe son del formato NetCDF, los cuales pueden ser leídos por herramientas externas como por ejemplo Matlab, que es el caso para la realización de esta aplicación meteorológica.[19]

### 2.4.3. Creación de un dominio de estudio para WRF

Mediante una herramienta gráfica se puede crear el dominio de estudio y generar los archivos NetCDF para su posterior utilización, este programa se llama “WRF Domain Wizard” el cual consta de 7 simples pasos los cuales se describen a continuación.

1. Al abrir el programa, se selecciona los directorios que ahí se señalan los cuales corresponden a la ubicación del programa WPS, la carpeta donde está alojada la información geográfica (descargada al momento de instalar el modelo) y por ultimo la dirección donde se encuentra la información de los dominios existentes (donde posteriormente se guardará el dominio que se creará).



Figura 2.13: Configuración del dominio.

**Fuente:** Elaboración propia.

2. En la pantalla siguiente, se escoge la opción “New domain” (Nuevo dominio). Notar que también se puede abrir un dominio ya existente, es por esto que en la ventana anterior se debe indicar el directorio de los dominios.

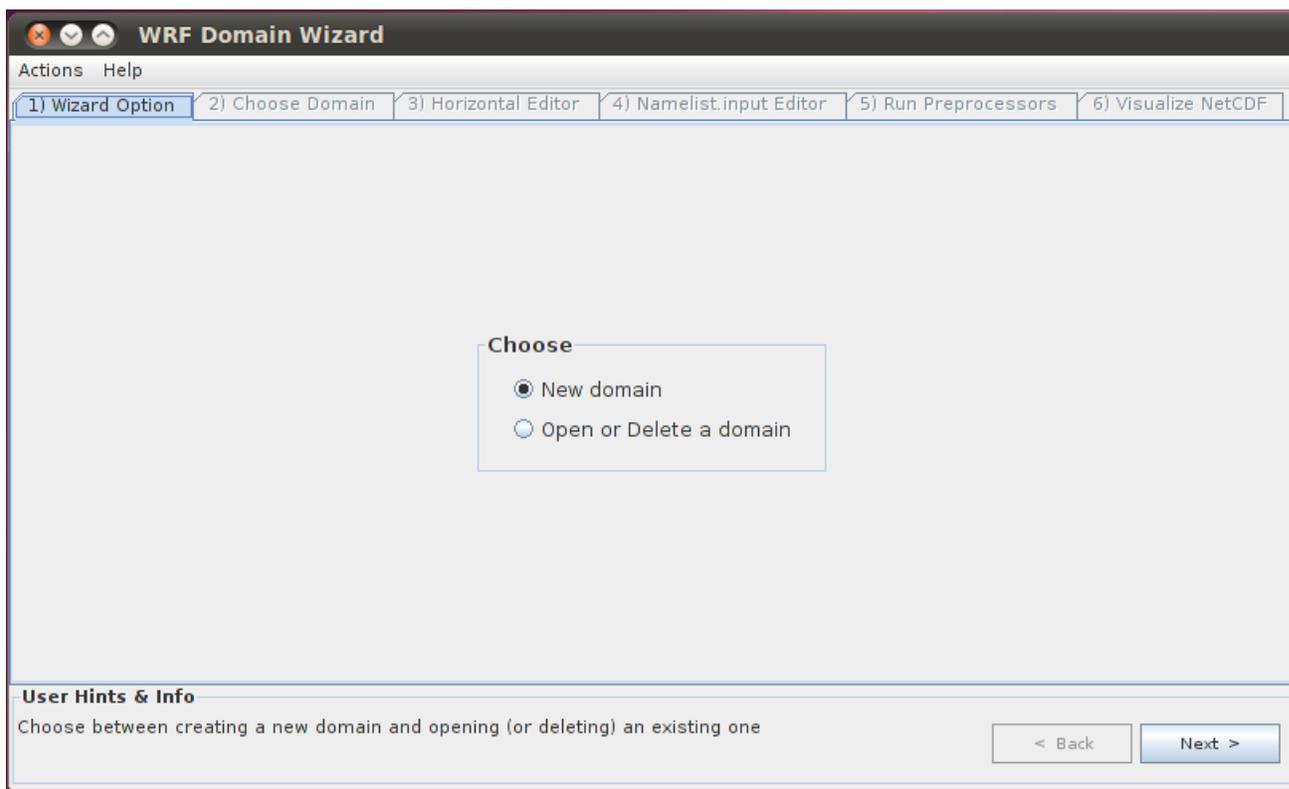


Figura 2.14: Opción de dominio.

**Fuente:** Elaboración propia.

3. A continuación se asigna un nombre y descripción al dominio que se desea crear. Para este ejemplo el nombre del dominio es “Chile” y su descripción “Territorio Nacional”.

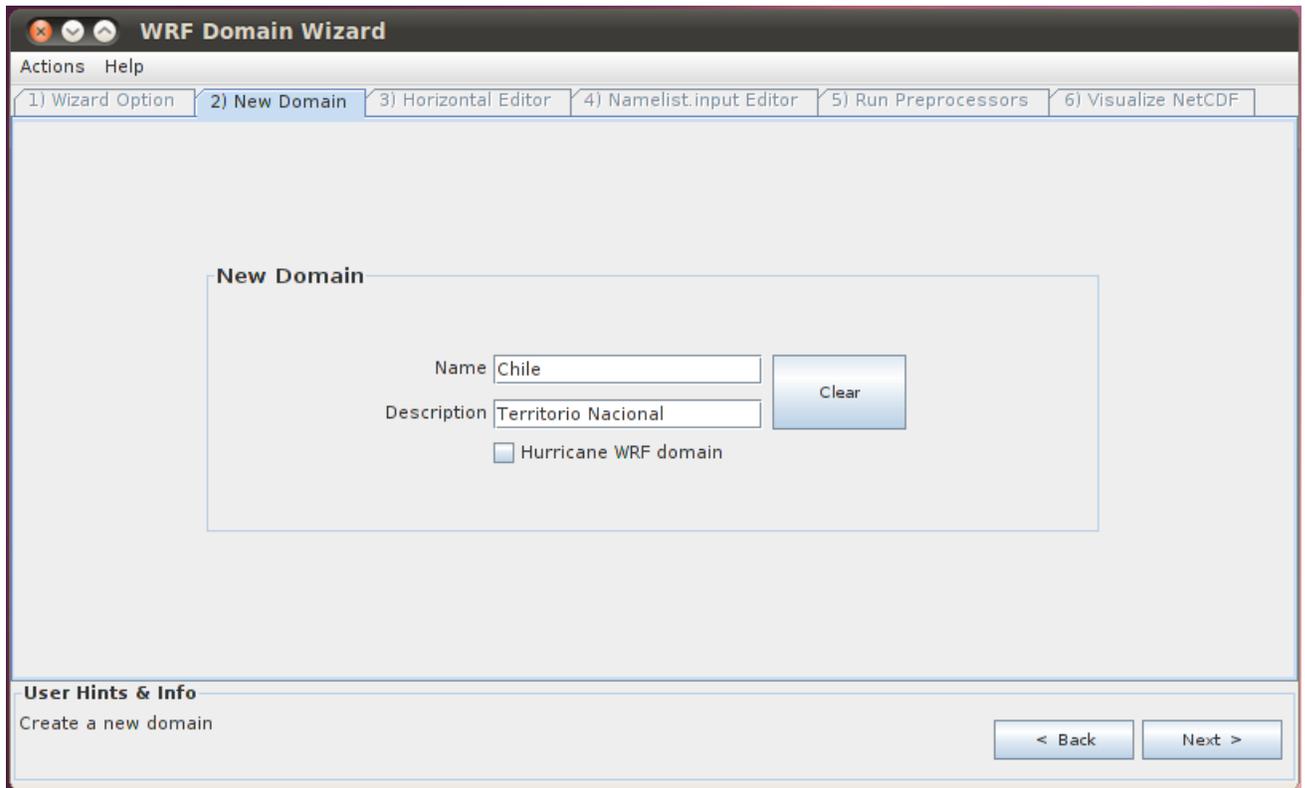


Figura 2.15: Definición de nombre y descripción del nuevo dominio.

**Fuente:** Elaboración propia.

- En esta etapa, se define el dominio propiamente tal, mediante la interfaz gráfica proporcionada por la herramienta, además de poder definir ciertos parámetros según desee el usuario del dominio.

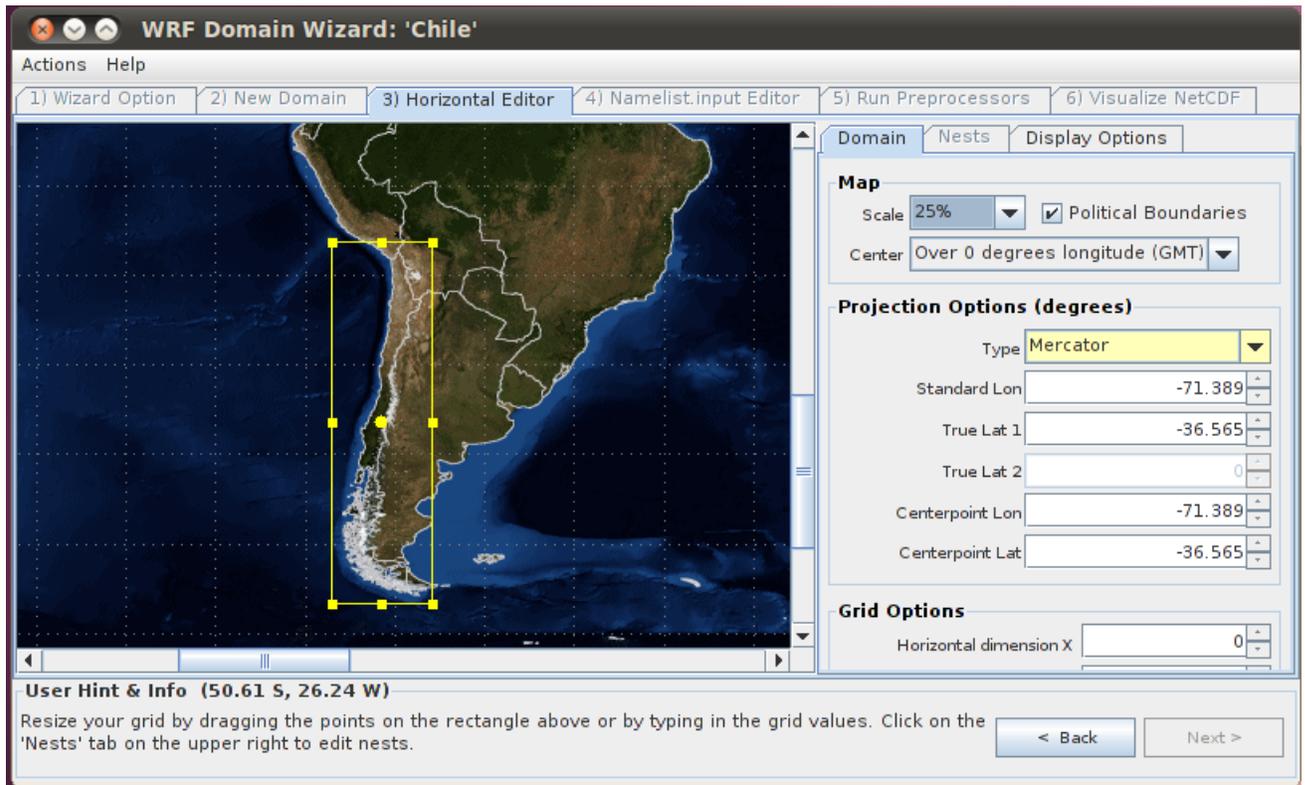


Figura 2.16: Definiición del dominio.

**Fuente:** Elaboración propia.

- Una vez creado el dominio, se presenta en pantalla todos los parámetros definidos en formato texto. Información que se almacena en un archivo denominado “namelist.input” el cual es utilizado posteriormente por los programas que comprenden el modelo.

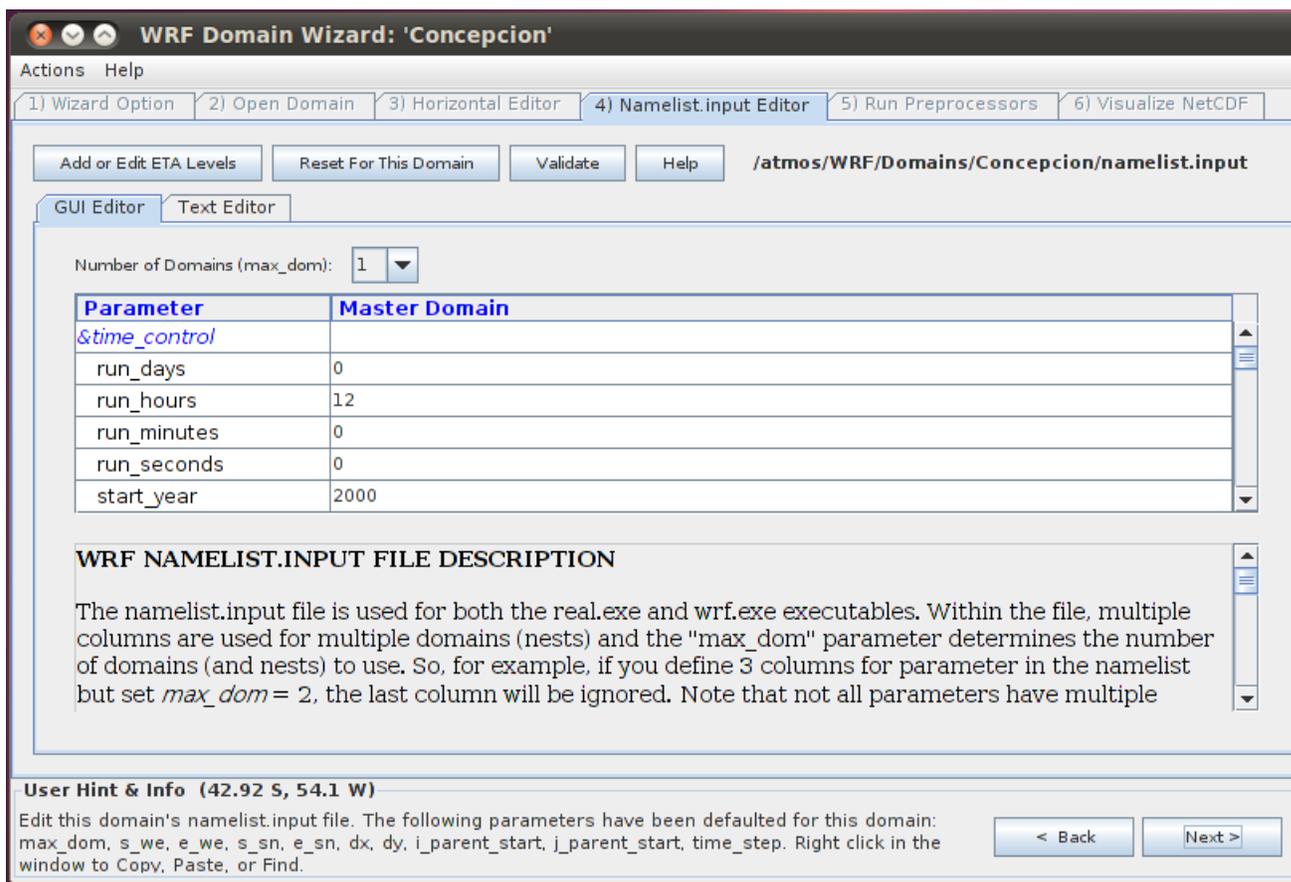


Figura 2.17: Descripción del archivo “namelist.input”.

**Fuente:** Elaboración propia.

- En la siguiente pantalla, se puede ejecutar de forma manual los programas: geogrid, ungrid y metgrid (definidos anteriormente). Cuando uno ejecuta cada uno de los procesos, puede seguir el estado de la ejecución en la pantalla de comando que se encuentra en la misma ventana.

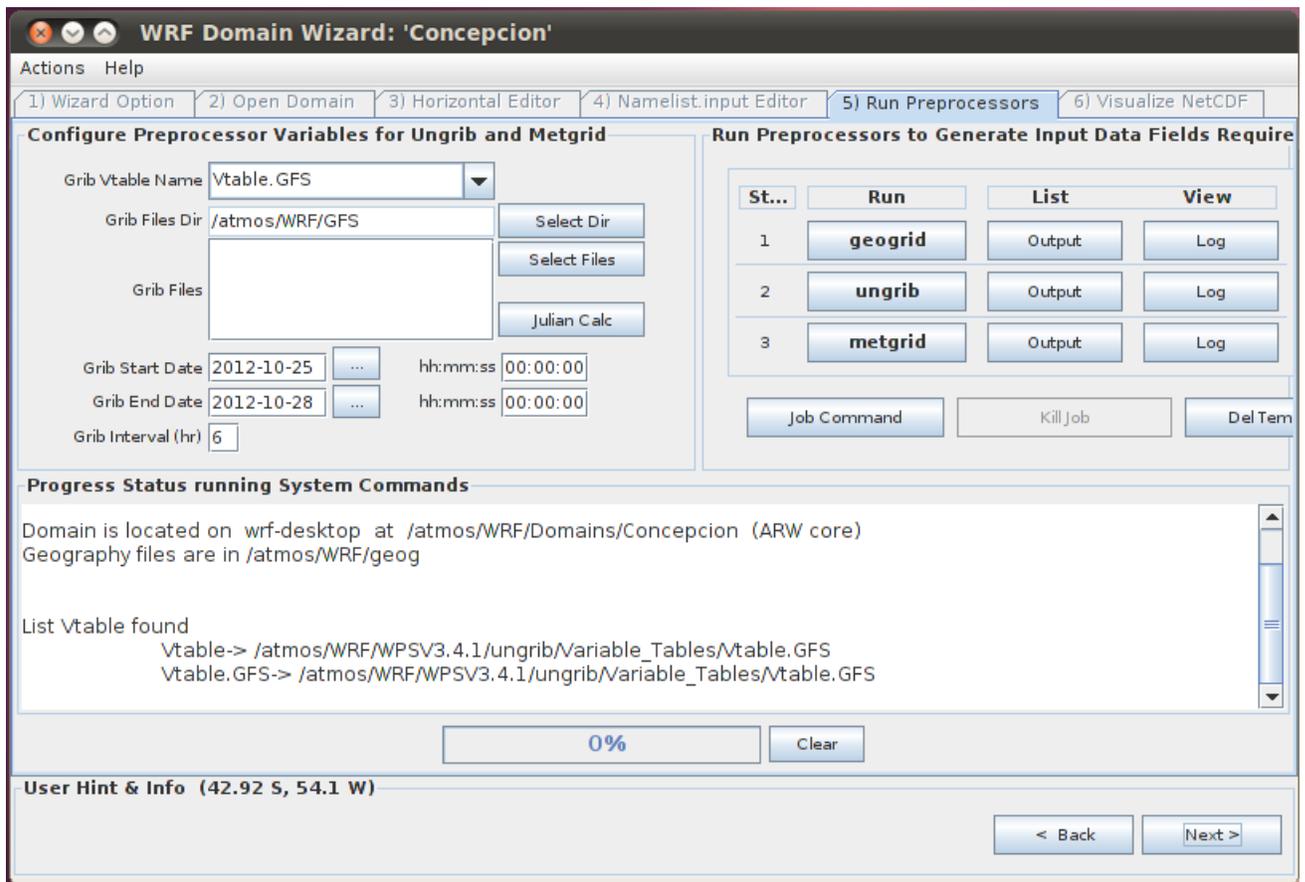


Figura 2.18: Pantalla de ejecución de Preprocesadores.

**Fuente:** Elaboración propia.

7. En la última etapa de la herramienta se aprecia la lista de archivos NetCDF creados por el modelo.

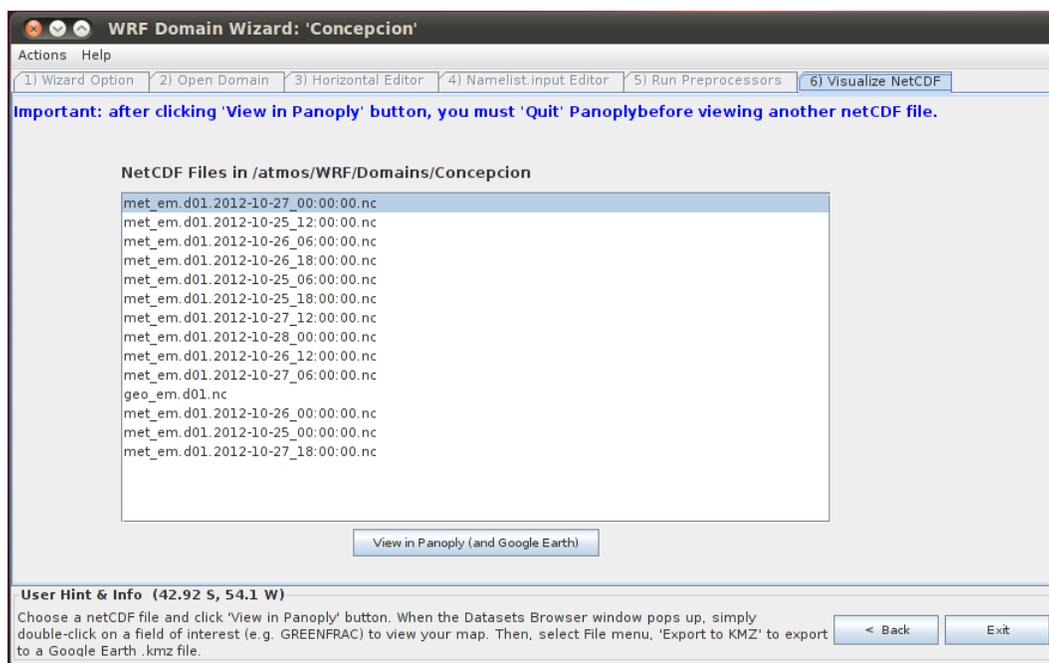


Figura 2.19: Archivos NetCDF creados.

**Fuente:** Elaboración propia.

## 2.5. MATLAB (MATrix LABoratory)

### 2.5.1. Introducción

MATLAB es una herramienta de software matemático que ofrece un entorno interactivo para el cálculo numérico y un lenguaje de programación propio (lenguaje M).

Mediante MATLAB, es posible analizar datos, desarrollar algoritmos y crear modelos o aplicaciones, tales como procesamiento de señales y comunicaciones, procesamiento de imagen y vídeo, sistemas de control, pruebas y medidas, finanzas computacionales y biología computacional. El lenguaje, las herramientas y las funciones matemáticas incorporadas permiten además explorar diversos enfoques.

MATLAB se puede utilizar en una gran variedad de aplicaciones, tales como procesamiento de señales y comunicaciones, procesamiento de imagen y vídeo, sistemas de control, pruebas y medidas, finanzas computacionales y biología computacional.

Se encuentra disponible para las plataformas GNU/Linux, Windows, Unix y Mac OS X.[20]

### 2.5.2. Características Principales

- Lenguaje de alto nivel para el cálculo numérico, la visualización y el desarrollo de aplicaciones.
- Entorno interactivo para la iterativa exploración, el diseño y la solución de problemas.
- Funciones matemáticas para álgebra lineal, estadística, análisis de Fourier, filtrado, optimización, integración numérica y resolución de ecuaciones diferenciales ordinarias.
- Gráficos integrados para visualizar datos y herramientas para crear gráficos personalizados.
- Herramientas de desarrollo para mejorar la calidad y el mantenimiento del código, así como para maximizar el rendimiento.
- Herramientas para crear aplicaciones con interfaces gráficas personalizadas.
- Funciones para integrar algoritmos basados en MATLAB con aplicaciones y lenguajes externos tales como C, Java, .NET y Microsoft Excel.[20]

## 2.6. Aplicaciones móviles similares

A continuación se nombran las aplicaciones meteorológicas para Android mas descargadas, ya sean de pago o gratuitas que ofrece la tienda virtual Google Play.

- **WeatherPro:**

WeatherPro para Android es una aplicación de pago que ofrece pronóstico para siete días e informes meteorológicos para más de dos millones de localizaciones. Incluye imágenes satelitales globales y de radar solo para Estados Unidos, Australia y gran parte de Europa. Sin embargo después de adquirirla se debe seguir pagando para tener todas las características.



Figura 2.20: WeatherPro.

Fuente: Tienda Google Play.

- **GO Weather Forecast & Widgets:**

GO Weather es aplicación gratuita que predice el clima en más de 100.000 lugares por todo el mundo, pero que no es tan exacta para entregar la temperatura.



Figura 2.21: Go Weather Forecast & Widgets.

**Fuente:** Tienda Google Play.

- **MeteoEarth:**

MeteoEarth es una aplicación de pago para tablets y celulares que ofrece la predicción del tiempo mundialmente. Para obtener todas las características de predicción se debe volver a pagar, además de la cuota anual exigida.

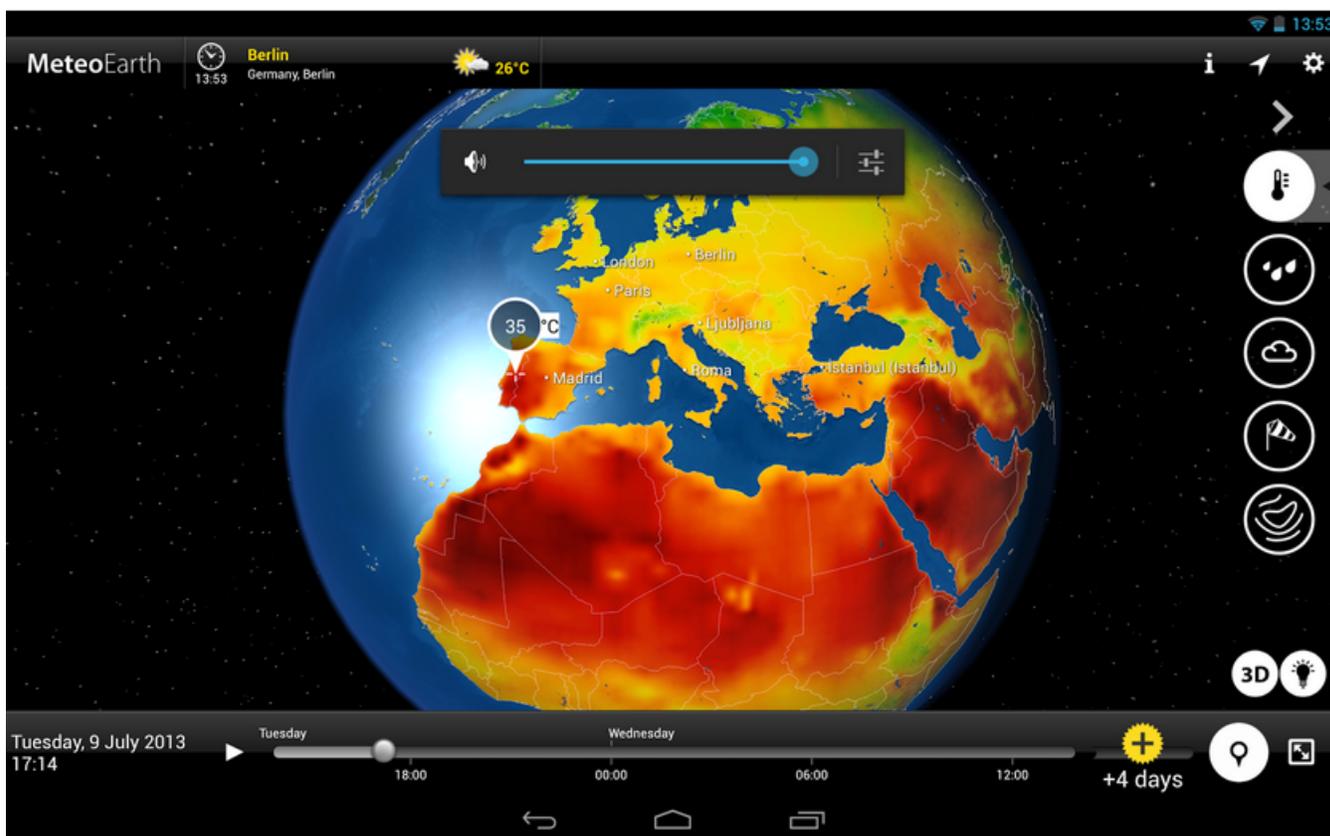


Figura 2.22: MeteoEarth.

Fuente: Tienda Google Play.

■ **AccuWeather Platinum:**

Aplicación de pago que ofrece los detalles de las condiciones meteorológicas locales. Esta aplicación proporciona predicciones por horas y envía alertas de condiciones adversas al dispositivo, solo para localidades de EE.UU.



Figura 2.23: AccuWeather.

**Fuente:** Tienda Google Play.

■ **eWeatherHD & Terremotos:**

EWeather HD produce un pronóstico de tiempo de 10 días con información barométrica, condiciones ambientales severas, pronósticos del tiempo y condiciones meteorológicas actuales. Proporciona información sobre los terremotos del mundo.



Figura 2.24: eWeatherHD & Terremotos.

**Fuente:** Tienda Google Play.

Todas estas aplicaciones tienen el mismo objetivo, que es desplegar por pantalla información climática de zonas geográficas, algunas con posibilidad de mostrar más días y otras con incluso información de terremotos, pero todas caen en el problema de la fidelidad de los datos, es decir que la predicción del tiempo posee un margen de error que la hace ser inexacta a la hora de entregar la temperatura, además para poseer todas las características que la aplicación dice que tiene se debe seguir pagando. Otras no son tan claras, ya que la pantalla se ve saturada por la cantidad de información volviéndola ilegible, como es el caso de eWeatherHD & Terremotos.

## Capítulo 3

# Definición Proyecto

### 3.1. Justificación Proyecto

El Departamento de Geofísica de la Universidad de Concepción realiza estudios sobre fenómenos climáticos, oceanográficos y sismológicos. En Chile los modelos para obtener pronósticos climáticos son muy generales considerando que por región presentan muy pocas localidades lo cual disminuye el grado de efectividad en el pronóstico, por ejemplo: que en la comuna de San Pedro de la Paz esté lloviendo no implica que en Coronel se dé el mismo caso.

Diariamente, a través de los modelos se obtienen varios Gigabytes de información con respecto al comportamiento del clima, el cual es aprovechado sólo por las personas que realizan estos análisis. La información que se genera es únicamente entendida por personas conocedoras de las herramientas que ocupan (imágenes 1.3, 1.4, 1.5, 1.6). Es por esto que se quiere llevar dicha información a la comunidad de la octava región de forma clara y entendible, la cual puede ser utilizada para diferentes fines, desde consultar para un paseo familiar, como también para determinar a priori el desarrollo del comportamiento del clima entre situaciones de riesgo (incendio, accidentes con materiales peligrosos, etc).

### 3.2. Objetivos del proyecto

- Objetivos generales:

Desarrollar una aplicación para dispositivos móviles (Android) con el fin de mostrar de forma clara y precisa los pronósticos climáticos que son entregados por los estudios realizados

en el departamento de Geofísica de la Universidad de Concepción para la región del Biobío.

■ **Objetivos específicos:**

- Investigar la plataforma Android para el desarrollo de aplicaciones móviles y aplicar los conocimientos adquiridos en el desarrollo de la aplicación final.
- Adaptar los datos entregados por el modelo de simulación de tal forma que el usuario pueda entenderlos mediante el dispositivo móvil.
- Diseñar y aplicar encuesta a potenciales usuarios de la aplicación para determinar requerimientos.
- Construir una aplicación de muestre la información de una forma clara y amigable para el usuario.

### **3.3. Ambiente de ingeniería de software**

#### **3.3.1. Metodología de desarrollo**

El equipo de desarrollo considera que una metodología ágil es la más propicia para implementar debido a sus múltiples ventajas por sobre las metodologías convencionales o tradicionales, ya que en su mayoría están preparadas para cambios durante el proyecto, los procesos son menos controlados y el cliente es una parte clave del trabajo en equipo, además, al ser un grupo de dos alumnos memoristas trabajando en el mismo sitio y al mismo tiempo, por lo que es más práctico utilizar una metodología ágil a una tradicional que tiende a ser compuesta por más integrantes con una cierta resistencia al cambio. [13]

Dentro de todas las metodologías ágiles se ha escogido la que se adapta más al contexto de este proyecto (recursos técnicos y humanos, tiempo de desarrollo, tipo de sistema, etc.-), y es la metodología XP (Programación Extrema) que reúne las mejores prácticas de las metodologías ágiles ya que es fundamental el trabajo en equipo y la retroalimentación ya que el código es revisado y es discutido mientras se escribe por lo que el equipo desarrollador debe estar en conocimiento de todos los aspectos del desarrollo del proyecto.

Otra razón para escoger XP es la falta de experiencia en las tecnologías necesarias para desarrollar el sistema, por lo que se pueden perfeccionar las habilidades a medida que se avanza en el proyecto.

### 3.3.2. Técnicas y notaciones

- Se realizará un modelo UML de diagrama “Casos de Uso”.

### 3.3.3. Estándares de documentación

Para el desarrollo del presente documento se utiliza la plantilla de documentación de proyecto de desarrollo de software de la Facultad de Ciencias Empresariales, Universidad del Bío Bío, versión del 3 de Noviembre del 2011. Esta plantilla es una adaptación basada en los siguientes estándares:

- IEEE Software Test Documentation Std 829 - 1998.
- IEEE Software Requirements Specifications Std 830 – 1998.

### 3.3.4. Herramientas de apoyo al desarrollo

- Tecnología Java, API de Android, PHP, XML.
- Entorno de desarrollo Eclipse con el plugin ADT.
- Servidor GNU/LINUX con Apache.
- Power Designer para modelamiento en UML.
- Matlab para la interacción con WRF.

### 3.4. Definiciones, siglas y abreviaciones

1. **App Store:** Es un servicio hecho para los dispositivos creados por Apple, que permite a los usuarios buscar y descargar aplicaciones. [8]
2. **Google play:** Es la plataforma para la descarga de aplicaciones para el sistema operativo Android. [21]
3. **UML:** El Lenguaje Unificado de Modelado, provee un estándar para realizar modelos en un lenguaje gráfico con el cual permite construir, documentar, visualizar y especificar un sistema de software. [22]
4. **API:** Significa “Interfaz de Programación de Aplicaciones”. Es una librería que nos proporciona funciones y procedimientos, o métodos si se trata de un lenguaje de programación orientado a objetos (POO) y los cual nos permite acceder a funciones que tienen otras aplicaciones. [35]
5. **XML:** Significa “Lenguaje de Marcado eXtensible”. Es una especificación/lenguaje que permite el intercambio de información estructurada entre diferentes plataformas, permitiendo la definición, transmisión, validación e interpretación de datos entre aplicaciones. [31]
6. **JDK:** Significa “Kit de Desarrollo Java”. Es un conjunto de programas y librerías para el desarrollo de aplicaciones en Java. [30]
7. **IDE:** Significa “Entorno de Desarrollo Integrado”. Es un programa que básicamente nos provee de una interfaz gráfica para programar, que corrige la sintaxis y compila el código de ser necesario, el cual puede estar enfocado en uno o más lenguajes de programación. [36]
8. **Plugin ADT:** Significa “Herramientas de Desarrollo en Android” (ADT). Es un plugin (programa a menor escala que se añade a otro) del IDE Eclipse que está diseñado para

- darle un ambiente potente e integrado en el cual construir aplicaciones Android. [9]
9. **SDK:** Significa “Kit de Desarrollo de Software”. Es un conjunto de herramientas que puede traer un depurador, compilador, ejemplos y documentación entre otras cosas. [11]
  10. **AVD:** Significa “Dispositivo Android Virtual”. Es una configuración de emulador que permite modelar un dispositivo Android mediante la definición de hardware y software. [9]
  11. **APK:** Es la extensión de un archivo, el cual viene empaquetado y se usa para distribuir e instalar aplicaciones para la plataforma Android. [9]
  12. **PHP:** Lenguaje de programación interpretado que funciona en el lado del servidor, diseñado originalmente para la creación de páginas web dinámicas y que se puede incluir directamente en el archivo HTML. [29]
  13. **Java:** Es un lenguaje de programación orientado a objetos, el cual adquiere gran parte de su sintaxis de C y C++. [28]
  14. **GUI:** Significa “Interfaz Gráfica de Usuario”. Es una interfaz visual que permite la interacción del usuario con una aplicación por medio de textos, botones, menús, imágenes, formularios, iconos, etc. [37]
  15. **GPS:** Significa “Sistema de Posicionamiento Global”. Es un sistema que permite determinar la posición geográfica con gran precisión por medio de un sensor GPS y 24 satélites que orbitan la tierra. [38]
  16. **IEEE:** Significa “Instituto de Ingenieros Eléctricos y Electrónicos”. Es una asociación técnica y profesional sin fines de lucro más grande del mundo dedicada a la estandarización, al avance de la innovación tecnológica, entre otras cosas. [27]

17. **ISO:** Significa “Organización Internacional de Normalización”. Es una institución que está encargada de especificar las normas a seguir en la creación de productos y servicios. [26]
18. **WRF:** Del Inglés “Weather Research and Forecasting Model” y su traducción al español sería “Modelo de investigación y pronósticos climáticos”, permite realizar estudios y simulaciones de diversos factores atmosféricos, oceanográficos, etc. [25]
19. **GNU/LINUX:** Sistema Operativo de software libre, por ende, de libre acceso y distribución. [39]
20. **Apache:** Uno de los servidores web más utilizados, de código abierto y uso gratuito. [34]
21. **Smartphone (Teléfono inteligente):** Es un teléfono móvil construido sobre una plataforma móvil con capacidad para realizar tareas semejantes a un mini computador. [40]
22. **Mac OS X:** Sistema operativo desarrollado, comercializado y vendido por Apple, que está incluido en el equipo de dicha compañía. [33]
23. **Widgets:** Vistas de miniaturas de las aplicaciones que pueden ser incrustadas en la pantalla inicio y recibir actualizaciones periódicas. [9]
24. **NFC(Near field communication):** Tecnología de comunicación inalámbrica, de corto alcance y alta frecuencia que permite el intercambio de datos entre dispositivos. [32]
25. **IDC (International Data Corporation):** Empresa estadounidense que realiza investigación de mercado, análisis y asesoramiento en todo el mundo. Especializada en tecnologías de la información , telecomunicaciones y tecnología de consumo . [23]

## Capítulo 4

# Especificación de requerimientos de software

Adaptación basada en *IEEE Software requirements Specifications* Std 830-1998.

### 4.1. Alcances

La aplicación tiene el propósito de comunicar al usuario con la información climática del modelo WRF, como lo son la temperatura, humedad, vientos y pronóstico de precipitaciones. Modelo que es utilizado por el departamento de geofísica de la Universidad de Concepción.

Esta versión contemplará solo la información correspondiente a las 54 comunas pertenecientes a la octava región.

La aplicación mostrará la información correspondiente a la ubicación del usuario mediante la geolocalización (si corresponde), y si así lo desea puede consultar el pronóstico para otra comuna revisando el mapa proporcionado por la misma aplicación mediante la API de Google Maps.

Este sistema completo está compuesto por 3 partes, las que son el modelo WRF que por medio de la creación de rutinas desarrolladas en matlab, se extrae la información relevante y la almacena en archivos, los cuales son interpretados por el lenguaje de programación web quien consulta estos datos según la solicitud realizada por el dispositivo Android, y entregará los datos necesarios para que la aplicación móvil despliegue la información de manera amigable al usuario.

## 4.2. Objetivos del software

### 4.2.1. Objetivos Generales

El sistema gestionará información sobre el clima en la región del Biobío, por lo que se le entregará al usuario el estado climático de la comuna que desee para un día completo o un resumen de tres días, por medio de la conexión del dispositivo Android con el servidor que recibirá y gestionará las peticiones correspondientes.

### 4.2.2. Objetivos Específicos

- Identificar la comuna en la que se encuentra el usuario, dentro de la región del Biobío.
- Por medio del mapa previsto por la aplicación el usuario podrá seleccionar cualquier comuna que desee dentro de la región del Biobío, para posteriormente consultar el pronóstico.
- Permitir la selección entre el pronóstico de un día o el de tres días.

## 4.3. Descripción global del producto

### 4.3.1. Interfaz de Usuario

- Para el correcto funcionamiento de la aplicación se requerirá que el dispositivo esté conectado a internet y el sensor GPS activado para poder trabajar de forma correcta.
- Para cada vista dentro de la aplicación la tecla de menú de Android desplegará un conjunto de acciones posibles, determinadas según la pantalla en la que se encuentra interactuando el usuario.
- Estará permitido visualizar la aplicación en las orientaciones portrait (vertical) o landscape (horizontal) según lo desee el usuario.

### 4.3.2. Interfaz de Hardware

- **Dispositivos Android:**

- Localizador GPS (Sistema de Posicionamiento Global).
- Conectividad Wifi o internet móvil mediante tecnología GPRS, 3G, 4G, etc.-

### 4.3.3. Interfaz de Software

- **Aplicación Android:**

- **Nombre:** Sistema Operativo Android.
- **Abreviación:** SO Android.
- **Versión:** Android 2.2 (Nombre de versión: Froyo).

- **Sistema Web:**

- **Nombre:** PHP.
- **Versión:** PHP 4 o superior.

- **Interacción con Modelo WRF:**

- **Nombre:** Matlab.
- **Abreviación:** Matlab es la abreviatura de MATrix LABoratory.
- **Versión:** Matlab 7 o superior.

### 4.3.4. Interfaz de Comunicación

Para la utilización del GPS se emplea el protocolo NMEA 0183 (de forma abreviada simplemente NMEA), y para la comunicación a través de internet se requiere del protocolo TCP/IP.

## 4.4. Requerimientos específicos

### 4.4.1. Requerimientos funcionales del sistema

Id	Nombre	Descripción
RF_01	Visualización de Mapa	Mostrar un mapa de navegación estilo Google Maps, en el cual se pueda visualizar el nombre de las calles, sentido, comunas, estacionamientos, entre otras capas de apoyo cartográfico.
RF_02	Gestión posición actual	Indicar la posición del usuario a través del sensor GPS del dispositivo Android.
RF_04	Control de comunicación	Alertar al usuario cuando no esté establecida la conexión a internet.
RF_05	Visualización de información	Desplegar correctamente la información que se ha solicitado, según la comuna y días de pronóstico.
RF_06	Gestión de archivos	Creación y lectura de los archivos.

Tabla 4.1: Requerimientos funcionales.

**Fuente:** Elaboración propia.

#### 4.4.2. Interfaces externas de entrada

Id	Nombre del ítem	Detalle de datos contenidos en ítem
DE_01	Datos Climáticos	Temperatura, variables de viento, presión atmosférica, variables de lluvia.

Tabla 4.2: Interfaces externas de entrada.

**Fuente:** Elaboración propia.

#### 4.4.3. Interfaces externas de salida

Id	Nombre del ítem	Detalle de datos contenidos en ítem	Medio de salida
IS_01	Datos clima para un día	Figura indicando estado climático (soleado, nublado, lluvioso, etc.-), temperatura, presión atmosférica, velocidad del viento y dirección, porcentaje de humedad.	Pantalla
IS_02	Datos clima para más días	Figura indicando estado climático (soleado, nublado, lluvioso, etc.-), temperatura mínima y máxima, presión atmosférica promedio, velocidad del viento promedio y dirección, porcentaje de humedad promedio.	Pantalla

Tabla 4.3: Interfaces externas de salida.

**Fuente:** Elaboración propia.

#### 4.4.4. Atributos del producto

Para el desarrollo del software se ha enfatizado en la usabilidad-operabilidad porque la aplicación debe ser una ayuda al usuario, es por esto que la interfaz debe ser muy amigable lo que se traduce en un uso muy intuitivo sin limitantes que entorpezcan la experiencia del usuario. Por

otro lado se encuentra la fiabilidad porque la información a la que accede la aplicación es proporcionada por el modelo WRF, modelo que como ya se ha mencionado anteriormente está avalado por organismos internacionales.

- **Funcionalidad:** La aplicación busca cumplir con las necesidades del usuario y los requerimientos establecidos por el cliente y potenciales usuarios que fueron consultados mediante una encuesta.
  - **Adecuación:** El sistema se adapta a los requerimientos que se especifican.
  - **Exactitud:** El sistema es capaz de entregar la información precisa en cuanto a la información climática del modelo de donde se extraen los datos.
  - **Inoperabilidad:** Esto ocurre única y exclusivamente cuando el dispositivo no tenga conexión a internet.
  - **Complacencia:** El producto se adhiere a los estándares en cuanto a la funcionalidad, y al ser muy intuitivo satisface las necesidades del usuario.
  - **Seguridad:** El sistema al no poseer datos de cuentas de usuarios o algún tipo de información sensible, se obvia este atributo.
- **Fiabilidad:** El sistema mantiene un alto grado de confiabilidad en el funcionamiento del modelo en el cual se respalda, lo que significa un porcentaje mínimo en cuanto a información errónea que se pueda entregar al usuario.
  - **Madurez:** El sistema está desarrollado para evitar errores en el funcionamiento del software.
  - **Tolerancia a las fallas:** La aplicación es altamente flexible a las fallas ya que se mantiene funcionando en condiciones óptimas.

- 
- **Recuperabilidad:** La capacidad del producto para recuperar información es óptima ya que el modelo está constantemente generando nuevos pronósticos.
  
  - **Usabilidad:** La aplicación ha sido desarrollada para ser muy fácil de usar con una interfaz sencilla e intuitiva.
    - **Comprensión:** La aplicación se comprende de manera intuitiva, dada las opciones que tiene el usuario para interactuar.
  
    - **Facilidad de aprendizaje:** Como se ha mencionado es muy simple de utilizar, pero de todos modos se incorpora una opción en el menú para ver las instrucciones de funcionamiento.
  
    - **Operabilidad:** Las operaciones que puede realizar el usuario dentro de la aplicación son fáciles de asimilar, para que el usuario obtenga la información que necesita.
  
  - **Mantenibilidad:** El sistema está altamente comentado, para que sea más fácil realizar posibles adaptaciones y/o mejoras.
    - **Capacidad De ser analizado:** por lo comentado del código, es sencillo y factible analizar el sistema.
  
    - **Capacidad De cambio:** de requerirse alguna modificación, la codificación del sistema lo permite por ser detallada.
  
    - **Estabilidad:** Dada la codificación de la aplicación, el software no debe presentar efectos inesperados de ningún tipo.
  
  - **Portabilidad:** La aplicación solo podrá funcionar en dispositivos con sistema operativo Android en las versiones 2.2 o superior.

- 
- **Adaptabilidad:** Dentro de las versiones de Android funcionará sin problemas. Portarlo a otro sistema operativo que no sea Android, será prácticamente imposible.
  - **Facilidad de Instalación:** Las formas de instalar la aplicación son variadas, una vez que la aplicación esté en la tienda de aplicaciones de Google “Google Play” el usuario podrá descargar desde la página web de la tienda la aplicación o desde la aplicación de la tienda que existe para dispositivos android también llamada “Google Play”, otra alternativa será poseer el programa de instalación dentro del dispositivo, se distingue el archivo por poseer la extensión .apk lo que indica que es el instalador de una aplicación de Android.
  - **Coexistencia:** La aplicación no altera el funcionamiento de otras tareas o programas instalados, solo depende de la conexión a internet y el servicio de GPS para operar.
  - **Reemplazabilidad:** Si bien, existen aplicaciones que también presentan información climática, los datos no son tan confiables como los del modelo WRF que son utilizados por la aplicación que se presenta.

## Capítulo 5

# Estudio de factibilidad

### 5.1. Introducción

Este apartado es fundamental para el desarrollo de cualquier proyecto de software por lo que se debe analizar cuidadosamente para no fallar ni comprometer el término y éxito del proyecto.

Se especifican los dispositivos a utilizar para el desarrollo y operación del sistema, considerando el software y hardware necesario mínimo para llevar a cabo las tareas.

También se contempla el ámbito económico en cuanto a los costos y beneficios que se puedan generar para el desarrollo del proyecto.

### 5.2. Factibilidad técnica

A continuación se detallan el equipamiento y software necesario para el desarrollo y funcionamiento del sistema.

### 5.2.1. Software utilizado para el desarrollo

Categoría	Software	Licencia
Plataforma	Windows 7 Ultimate Android 2.2	Microsoft CLUF Apache 2.0 y GNU GPL
Editor de código	Eclipse Kepler Sublime Text 3	Licencia pública de Eclipse Licencia propietaria
Servidor local (Wampserver)	Apache 2.4.4 PHP 5.4.12	Licencia Apache 2.0 Licencia PHP (Open Source)
Navegador web	Google Chrome 31 Firefox 23.0.1	Freeware MPL
Modelado	PowerDesigner 16.1	Licencia Propietaria

Tabla 5.1: Software para el desarrollo.

**Fuente:** Elaboración propia.

### 5.2.2. Hardware mínimo para el desarrollo

- Procesador Pentium IV 1.8Ghz.
- Memoria RAM 1Gb.
- Disco Duro 40 Gb.
- Tarjeta de Red.

- Tarjeta de Video 128 Mb.
- Mouse.
- Teclado.

### 5.2.3. Software mínimo para el funcionamiento

Sistema operativo	Licencia
Android 2.2 o superior.	Apache 2.0 y GNU GPL.

Tabla 5.2: Software mínimo para el usuario.

**Fuente:** Elaboración propia.

Programa	Licencia
Sistema operativo Linux	GNU
Servidor HTTP Apache 2.2.21	Apache 2.0
PHP 4 o superior	Licencia PHP (Open Source)

Tabla 5.3: Software mínimo para el Servidor.

**Fuente:** Elaboración propia.

### 5.2.4. Hardware mínimo para el funcionamiento

En este caso sólo se considera al usuario, y este debe poseer un smartphone o tablet.

### 5.2.5. Conocimientos del equipo de trabajo

#### ■ Lenguajes de programación

- Java.
- XML.
- PHP.
- JSON.
- Matlab.

#### ■ Programas

- Eclipse Kepler.
- Sublime Text 3.
- PowerDesigner 16.1.
- Android SDK Manager.
- Android Virtual Device Manager.

#### ■ Sistema operativos

- Windows.
- Android.

### 5.2.6. Conclusión de factibilidad técnica

Al haber completado el análisis de la factibilidad técnica y en conjunto con la base de los elementos presentados para el desarrollo del proyecto en cuanto a hardware y software por parte del equipo desarrollador y del servidor, se puede concluir que si es factible en términos técnicos.

Cabe destacar, que el usuario debe contar con dispositivo con sistema operativo Android 2.2 o superior en su versión ya que en la encuesta se puede ver que la concentración en cuanto a versiones está en las mayores a la 2.2. Para poder hacer uso de la aplicación debe cumplirse este requisito, porque de lo contrario no podrá instalar la aplicación.

A pesar de que los lenguajes de programación y las herramientas a utilizar son variados el equipo presenta experiencia en varios de ellos por lo que le da una ventaja para afrontar el desarrollo del sistema.

## 5.3. Factibilidad operativa

Se establecen los impactos positivos y/o negativos relacionados con la aplicación .

### 5.3.1. Impactos positivos

- Disminuir la incertidumbre de los usuarios.
- Poseer información confiable por cada comuna de la región.
- Información clara y entendible.
- Portabilidad.
- Aplicación amigable para el usuario.

### 5.3.2. Impactos negativos

- Las personas que no posean un dispositivo Android no podrán disfrutar de los beneficios de la aplicación.

### 5.3.3. Conclusiones de factibilidad operativa

Al finalizar el análisis de la factibilidad operativa, en cuanto a los impactos positivos se destaca como uno de los más importantes el hecho de que la aplicación aportará información confiable de una fuente respaldada por el departamento de Geofísica de la Universidad de Concepción y utilizada mundialmente para la investigación de fenómenos atmosféricos.

## 5.4. Factibilidad económica

En este apartado, se especifica todo lo relacionado con inversiones y ganancias que podrían surgir a partir de este proyecto y posterior producto.

### 5.4.1. Inversión de tangibles

En cuanto a equipamiento se refiere, no se realiza ninguna inversión por parte del equipo desarrollador ya que ambos integrantes disponen de equipos portátiles aptos para cumplir con el desarrollo del proyecto, por lo que no se considerará ninguna inversión por parte de los alumnos memoristas.

Por otra parte se encuentra el cliente (Departamento de Geofísica, Udec), el departamento cuenta con servidor propio y operativo, lo que se traduce en otro ahorro ya que no hay que considerar ningún costo de almacenamiento o hosting.

### 5.4.2. Inversión de intangibles

Todo el software necesario para el desarrollo del proyecto, el equipo desarrollador ya lo posee, por lo que no supone una inversión por parte de los alumnos.

La inversión que si es necesaria es un requerimiento por parte de la tienda de aplicaciones “Google Play”, quien exige para poder publicar una aplicación en la tienda, que el titular de la aplicación posea una cuenta de correo electrónico de Google (servicio llamado Gmail), y además desembolsar una suma de dinero con lo que la cuenta vinculada a la tienda quedará con permisos de publicación (siempre y cuando cumpla con los requisitos pertinentes) de por vida, por lo que la inversión en cuanto a suscripción a la tienda se efectúa solo una vez. Como se puede apreciar

en la tabla que está a continuación, la suma exigida por Google asciende a los USD\$25.

Inversión	Monto Publicado	Monto en moneda local
Suscripción a "Google Play"	USD\$25	CLP \$14025

Tabla 5.4: Suscripción a Google Play.<sup>1</sup>

**Fuente:** Elaboración propia.

### 5.4.3. Inversión de recursos humanos

Considerando que solo trabajarán en el proyecto los alumnos memoristas, no se contemplará la contratación de terceros, por lo que no habría costos respectivos a ese ámbito. También, al ser los alumnos memoristas quienes trabajen en el proyecto con el fin de obtener el título profesional al que optan, no suponen un costo.

### 5.4.4. Conclusión de factibilidad económica

En términos económicos, el proyecto si es factible ya que es el cliente quien tiene que costear la suscripción a la tienda "Google Play", monto que no genera problemas en cuanto a las finanzas del departamento.

## 5.5. Conclusión del estudio de factibilidad

En base al análisis previamente realizado en cuanto a factibilidad técnica, operativa y económica se concluye que el proyecto es factible en su totalidad.

Considerando los recursos tecnológicos (hardware y software) que el equipo desarrollador posee, no hace falta adquirir ningún elemento adicional. También el gasto económico es mínimo por parte del departamento de Geofísica, ya que solo se debe considerar la suscripción vitalicia a Google Play, y en cuanto a equipamiento, el departamento posee servidor el cual está operativo, lo que se traduce en otro ahorro en cuanto a factibilidad operativa se refiere.

<sup>1</sup>Valor de moneda local está sujeto a cambios según variación del dólar, para este caso se considera USD\$1=CLP\$561,  $25 \times 561 = 14025$  (valor del dólar para el día 27 de febrero de 2014).

## Capítulo 6

# Análisis

A continuación se presenta el diagrama de casos de uso que tiene la finalidad de representar la forma en cómo los actores propuestos operan con el sistema.

Cada caso de uso es especificado considerando las condiciones previas para su funcionamiento y el flujo de eventos básicos.

### 6.1. Diagrama Caso de Uso

#### 6.1.1. Actores

- **Usuario**

- **Rol:** Es la persona que hace uso de la aplicación móvil, para saber el pronóstico del tiempo para la comuna de la octava región donde se encuentre, o de otra comuna que desee consultar.
- **Conocimientos técnicos:** Utilización de dispositivos Android.
- **Privilegios:** Determinar la comuna en la que se encuentra dentro de la octava región, navegar en el mapa, consultar el pronóstico para el mismo día o un resumen para 3 días en la comuna en la que se encuentra o en alguna otra si así lo prefiere.

- **Modelo WRF**

- **Rol:** Se encarga de generar la predicción numérica de los pronósticos climáticos.

- **Google Maps**

- **Rol:** Se encarga de la visualización y actualización del mapa que posee la aplicación.

- **GPS Android**

- **Rol:** Se encarga de obtener y gestionar la posición del dispositivo móvil.

### 6.1.2. Casos de Uso

En la Fig. 6.1 se expone el diagrama de casos de uso que representa las funcionalidades de sistema, y su interacción con los actores.

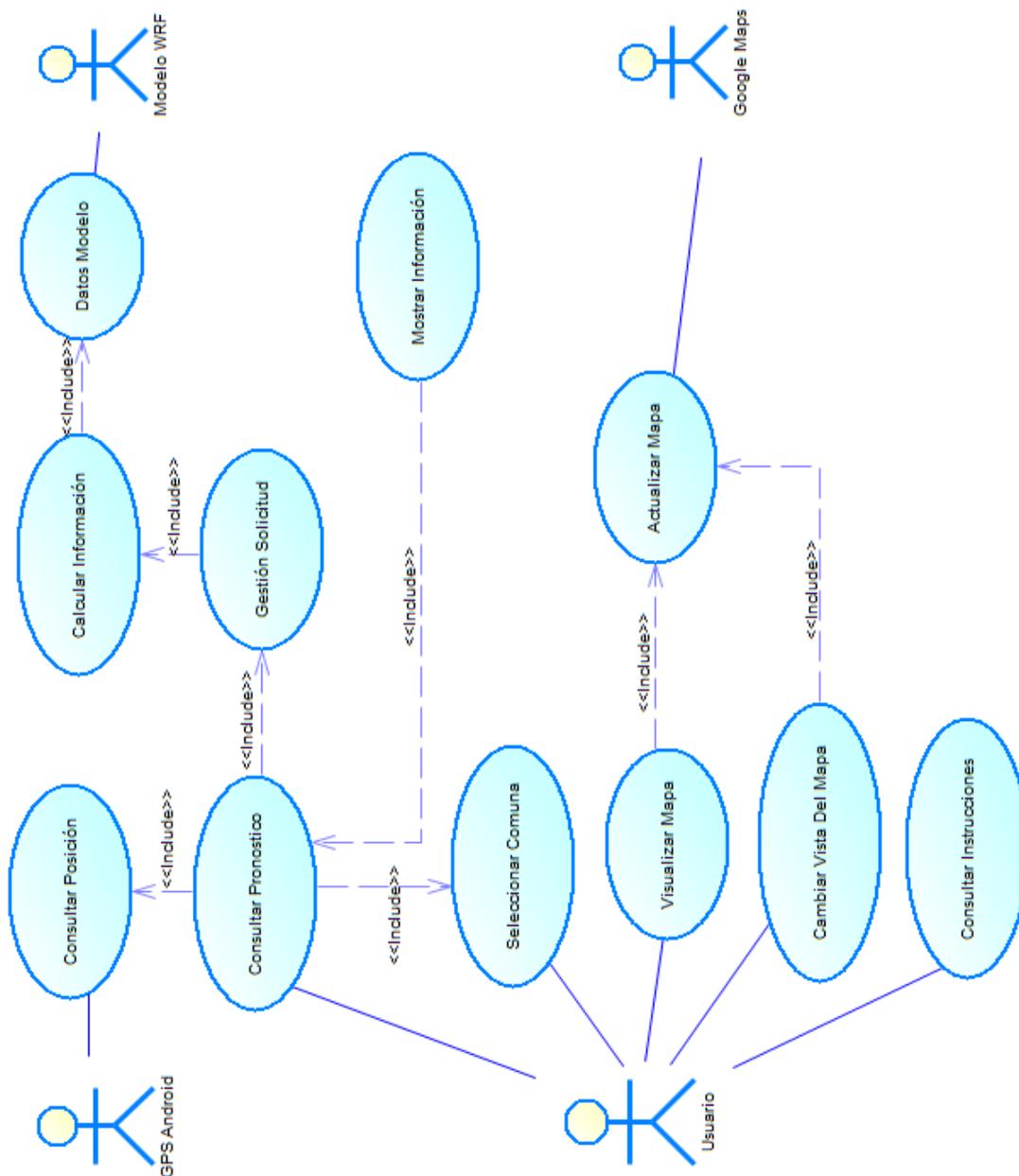


Figura 6.1: Diagrama casos de uso.

**Fuente:** Elaboración propia.

### 6.1.3. Descripción general del uso del sistema

El usuario abrirá la aplicación en su dispositivo móvil, y lo primero que le indicará ésta es en qué comuna se encuentra dentro de la octava región mediante el uso del GPS provisto por Android, en este momento el usuario tendrá la opción de consultar el tiempo para el mismo día o un pronóstico resumido para tres días.

Para poder acceder a la otra funcionalidad importante de la aplicación que es consultar por cualquier comuna de la región del Bío-bío, el usuario simplemente tiene que cambiar de pestaña presionando donde dice “Región del Bío-bío” donde se visualiza un mapa donde se puede ver completamente la octava región. En el mapa el usuario podrá pulsar el lugar que desee consultar, una vez que puse el mapa, aparecerá un marcador que indicará la comuna que la presionado, para así poder luego consultar el pronóstico del tiempo para el mismo día o para tres días.

Cuando el usuario visualiza el mapa, si presiona el botón de menú de su dispositivo Android, tendrá las opciones de ver instrucciones de uso, así como también una opción para cambiar la vista del mapa.

### 6.1.4. Especificación de los casos de uso

#### 1. Caso de Uso <Consultar Posición>:

- **Descripción:** Consulta la posición del sensor GPS.
- **Pre-Condición:**  
El dispositivo debe contar con un sensor GPS.  
El sensor GPS debe estar activado.
- **Flujo de eventos básicos:**

Actor	Sistema
1.- El dispositivo envía las coordenadas de donde se encuentra actualmente	2.- El sistema obtiene la posición actual del dispositivo con GPS

Tabla 6.1: Flujo de eventos básicos: <Consultar Posición>.

**Fuente:** Elaboración propia.

▪ **Flujo de eventos alternativos:**

Actor	Sistema
1.- El dispositivo se encuentra con el GPS desactivado por lo que muestra mensaje de alerta.	

Tabla 6.2: Flujo de enentos alternativos: <Consultar Posición>.

**Fuente:** Elaboración propia.

- **Post-condición:** Se obtiene la posición actual del dispositivo móvil. En caso de ocurrir el flujo alternativo, se le solicita al usuario activar las funcionalidades del sensor GPS y poseer una conexión a internet estable.

2. Caso de Uso <Datos Modelo>:

- **Descripción:** En este caso de uso se extrae la información necesaria del modelo WRF, almacenándola por cada comuna en un archivo.
- **Pre-Condición:** Para poder extraer los datos, es critico que se ejecute el modelo WRF previamente.
- **Flujo de eventos básicos:**

Actor	Sistema
1.- Se ejecuta el modelo WRF, finalizando con la creación de los archivos NetCDF.	2.- Las rutinas de Matlab extraen la información de los archivos NetCDF, creando los archivos correspondientes para cada comuna de la octava región.

Tabla 6.3: Flujo de eventos básicos: <Datos Modelo>.

**Fuente:** Elaboración propia.

- **Post-condición:** Una vez ya creados los archivos por comuna, la aplicación queda apta para consultar los datos actualizados.

3. Caso de Uso <Calcular Información>:

- **Descripción:** Según el tipo de pronóstico que se consulte (un día o más días), calcula la información a partir del archivo de la comuna donde se consulta. El calculo significa obtener temperaturas (del instante o un resumen con la mínima y máxima), calcular humedad, hacer los cálculos respectivos con las variables de viento para obtener la velocidad y dirección del mismo, además de determinar cual será la imagen que se desplegará según en análisis de las variables de lluvia y nubosidad.
- **Pre-Condición:** Debe existir una petición de pronóstico.
- **Flujo de eventos básicos:**

Actor	Sistema
	1.- Calcular los datos que serán posteriormente enviados al dispositivo. Los datos a calcular son: temperatura mínima y máxima, porcentaje de humedad, velocidad y dirección del viento, con las variables de lluvia y vapor de agua acumulada se determina que figura se muestra en el pronóstico.

Tabla 6.4: Flujo de eventos básicos: <Calcular Información>.

**Fuente:** Elaboración propia.

- **Post-condición:** La información está preparada para ser enviada al dispositivo.

4. Caso de Uso <Consultar Pronóstico>:

- **Descripción:** Es una acción primordial para el sistema, según la visualización que tenga el usuario en la aplicación, el caso de uso va a hacer uso de otros casos de uso relacionados para cumplir su objetivo que es dada una comuna de la octava región, desplegar por pantalla el pronóstico correspondiente.
- **Pre-Condición:** Al momento de abierta la aplicación, se debe contar con una conexión a internet y el sensor GPS debe estar activado.
- **Flujo de eventos básicos:**

Actor	Sistema
	1.- Al iniciar la aplicación, se utiliza el caso de uso "Consultar posición", para determinar la ubicación del usuario y determinar su comuna.
2.- El usuario determina que pronóstico desea, si para un día o para más días.	3.- Se utiliza el caso de uso "Gestión de solicitud".
	4.- Se utiliza el caso de uso "Mostrar información".

Tabla 6.5: Flujo de eventos básicos: <Consultar Pronóstico>.

**Fuente:** Elaboración propia.

- **Post-condición:** Se activa el caso de uso “Mostrar información”.

## 5. Caso de Uso &lt;Gestión Solicitud&gt;:

- **Descripción:** Se encarga de enviar la petición al servidor y posteriormente recibir la respuesta correspondiente.
- **Pre-Condición:** Se tienen que solicitar un pronóstico.
- **Flujo de eventos básicos:**

Actor	Sistema
1.- El usuario solicita un pronóstico.	2.- Se establece la conexión con el servidor, con los parámetros correspondientes.
	3.- Se utiliza el caso de uso "Calcular información".
	4.- La información referente al pronóstico se envía al dispositivo en el formato JSON.
	5.- la aplicación convierte la información recibida en formato JSON, a una lista objetos.

Tabla 6.6: Flujo de eventos básicos: &lt;Gestión Solicitud&gt;.

**Fuente:** Elaboración propia.

- **Post-condición:** La información necesaria ya está disponible por parte de la aplicación.

## 6. Caso de Uso &lt;Mostrar Información&gt;:

- **Descripción:** Se encarga de la visualización de la información.
- **Pre-Condición:** La aplicación debe recibir la información desde el servidor.
- **Flujo de eventos básicos:**

Actor	Sistema
	1.- Con la lista de objetos, se crea la vista para el usuario y se despliega en pantalla.
2.- El usuario puede visualizar el pronóstico.	

Tabla 6.7: Flujo de eventos básicos: &lt;Mostrar Información&gt;.

**Fuente:** Elaboración propia.

- **Post-condición:** El usuario puede ser el pronostico solicitado.

7. Caso de Uso <Seleccionar Comuna>:

- **Descripción:** En el mapa provisto por Google, se marca la comuna donde el usuario pulse la pantalla.
- **Pre-Condición:** El usuario debe habilitar la pestaña con el mapa de la región.
- **Flujo de eventos básicos:**

Actor	Sistema
1.- Usuario presiona pestaña del mapa de la región del Bio-Bio.	2.- La aplicación despliega el mapa.
3.- Usuario pulsa la pantalla.	4.- Aparece una marca en el mapa, sobre la comuna seleccionada.

Tabla 6.8: Flujo de eventos básicos: <Seleccionar Comuna>.

**Fuente:** Elaboración propia.

- **Post-condición:** el usuario puede solicitar un pronóstico en la comuna seleccionada.

8. Caso de Uso <Visualizar Mapa>:

- **Descripción:** Permite la visualización del mapa.
- **Pre-Condición:** Seleccionar la pestaña de “Región del Bío-Bío”.
- **Flujo de eventos básicos:**

Actor	Sistema
1.- El usuario pulsa la pestaña "Región del Bío-Bío."	2.- La aplicación muestra la visualización inicial del mapa, que comprende toda la región.
3.- el usuario puede desplazar el mapa o realiza una acción que requiere actualizar el mapa.	4.- El sistema invoca el caso de uso "Actualizar mapa" y obtiene la nueva representación geográfica.

Tabla 6.9: Flujo de eventos básicos: <Visualizar Mapa>.

**Fuente:** Elaboración propia.

- **Post-condición:** Se muestra la zona del mapa que corresponda.

9. Caso de Uso <Actualizar Mapa>:

- **Descripción:** Actualiza el área de mapa, como también se encarga de cambiar la forma de visualizar el mapa si el usuario así lo indica en las opciones del menú del mapa.
- **Pre-Condición:** Seleccionar la pestaña de “Región del Bio-Bío”.
- **Flujo de eventos básicos:**

Actor	Sistema
	1.- La aplicación detecta cambios en cuanto a la información que debe representar geográficamente.
2.- La API de Google Maps envía la nueva zona geográfica de la zona que se está desplegando en la aplicación.	3.- La aplicación muestra de forma correcta el mapa, según la interacción del usuario.

Tabla 6.10: Flujo de eventos básicos: <Actualizar Mapa>.

**Fuente:** Elaboración propia.

- **Post-condición:** Se actualiza el mapa.

10. Caso de Uso <Cambiar Vista Del Mapa>:

- **Descripción:** El usuario puede modificar la forma en que se visualiza el mapa.
- **Pre-Condición:** Seleccionar la pestaña de “Región del Bio-Bío”.
- **Flujo de eventos básicos:**

Actor	Sistema
1.- El usuario activa el cambio de vista.	2.- La aplicación cambia la apariencia del mapa, cambiando cada vez que se active el cambio de vista, entre las 4 apariencias disponibles.

Tabla 6.11: Flujo de eventos básicos: <Cambiar Vista Del Mapa>.

**Fuente:** Elaboración propia.

- **Post-condición:** Se actualiza el mapa con la nueva vista.

11. Caso de Uso <Consultar Instrucciones>:

- **Descripción:** Se despliega en pantalla todas las instrucciones para el buen uso de la aplicación.
- **Pre-Condición:** Ninguna.
- **Flujo de eventos básicos:**

Actor	Sistema
1.- El usuario consulta las instrucciones.	2.- La aplicación despliega en pantalla la lista de instrucciones para el buen uso de la aplicación.

Tabla 6.12: Flujo de eventos básicos: <Consultar Instrucciones>.

**Fuente:** Elaboración propia.

## Capítulo 7

# Diseño

### 7.1. Introducción

El presente capítulo está destinado a presentar el arquetipo de la aplicación móvil mediante diagramas de descomposición funcional en la sección de “Diseño de arquitectura funcional”.

También, en la sección “Diseño de interfaz y navegación” se presenta la iconografía utilizada, logo de la aplicación, además de las distintas vistas dentro de la aplicación.

Para finalizar se describen los módulos que componen a la aplicación.

## 7.2. Diseño de arquitectura funcional

En esta sección se plantean los procesos y módulos en términos generales mediante un diagrama de arquitectura funcional.

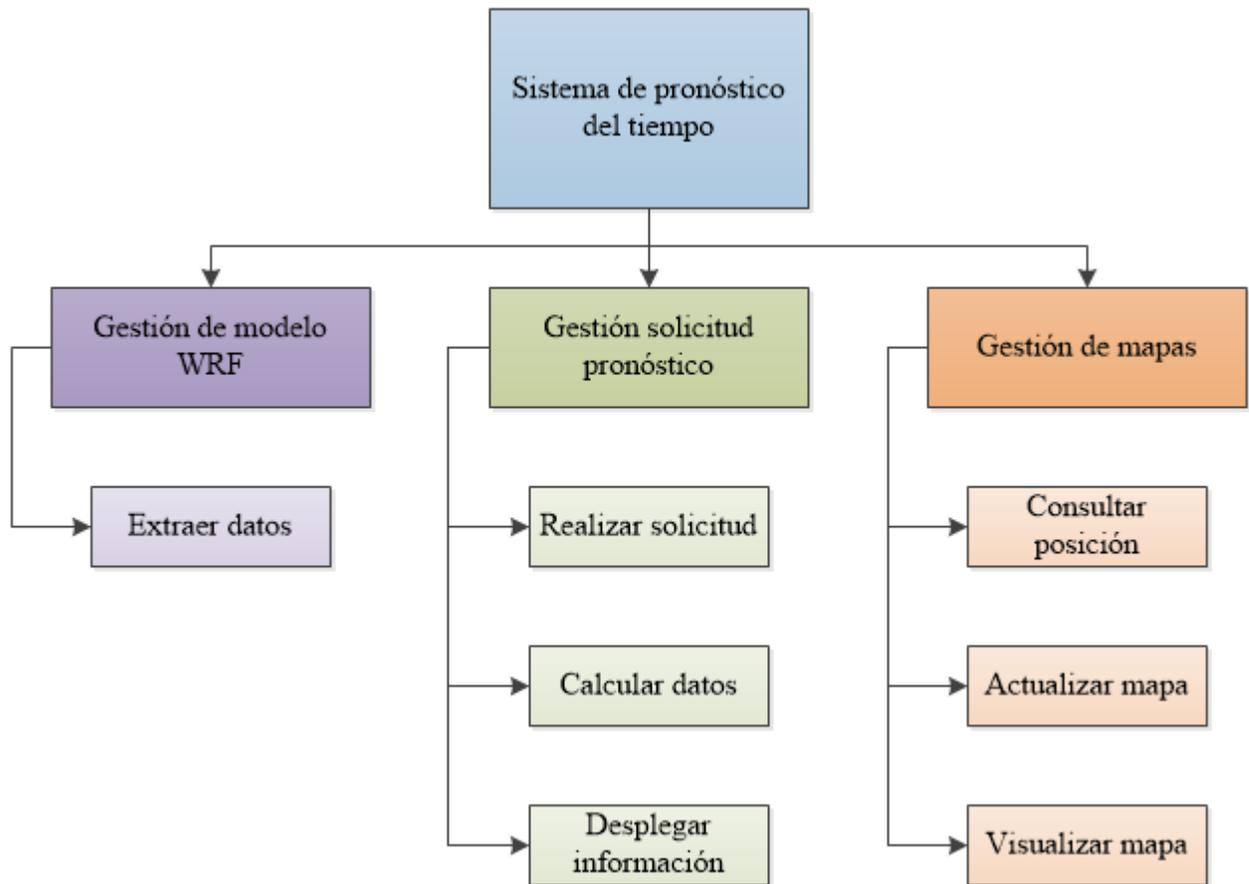


Figura 7.1: Diseño de arquitectura funcional - nivel general.

**Fuente:** Elaboración propia.

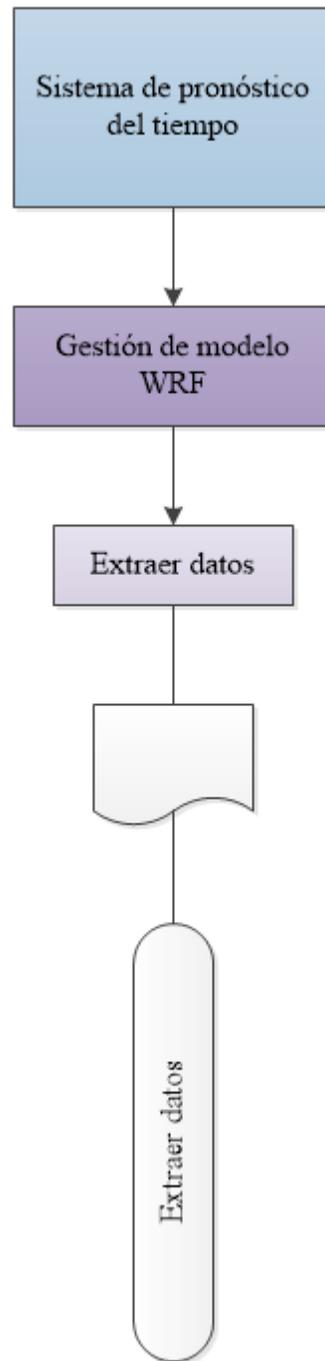


Figura 7.2: Descomposición funcional - Gestión de modelo WRF.

**Fuente:** Elaboración propia.

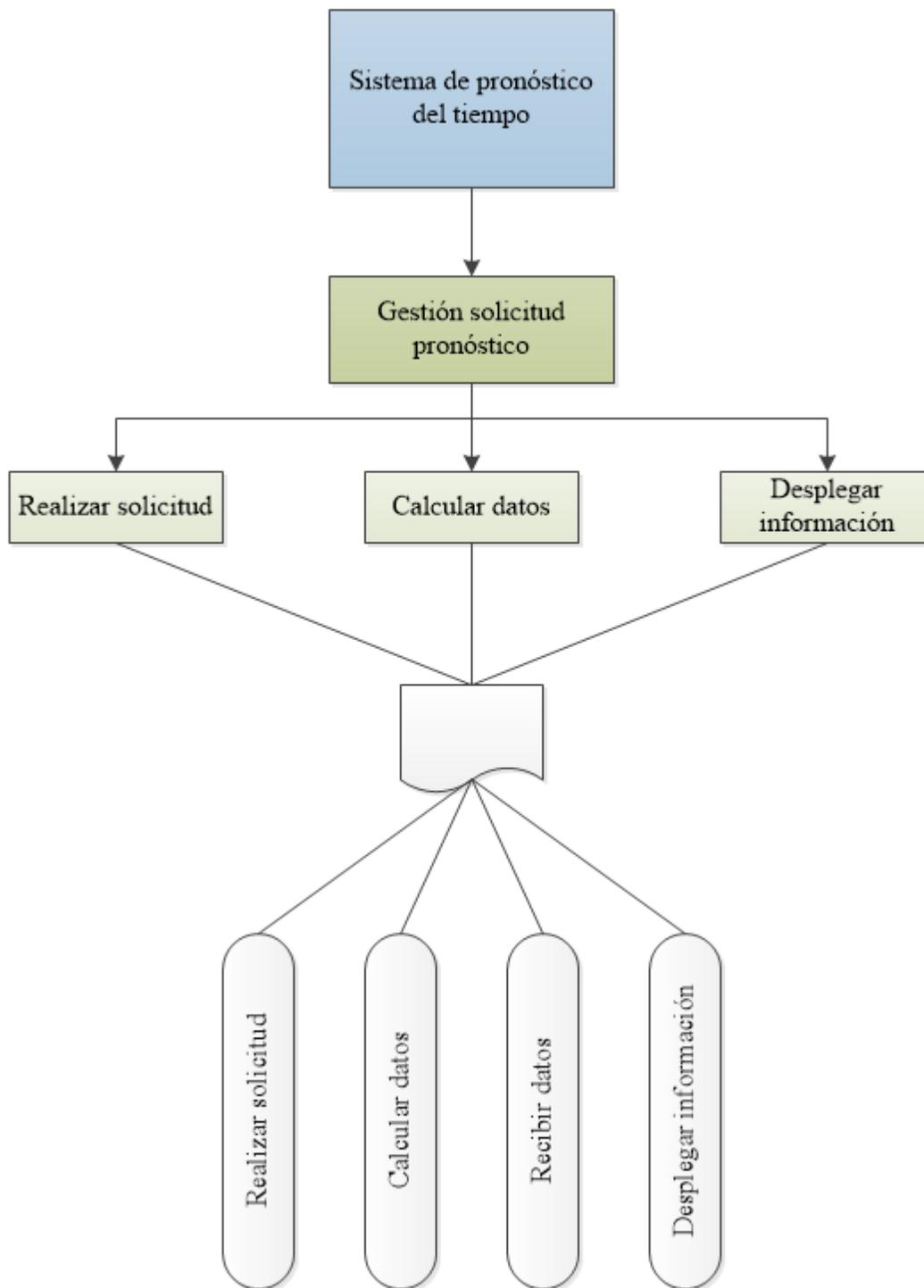


Figura 7.3: Descomposición funcional - Gestión solicitud pronóstico.

**Fuente:** Elaboración propia.

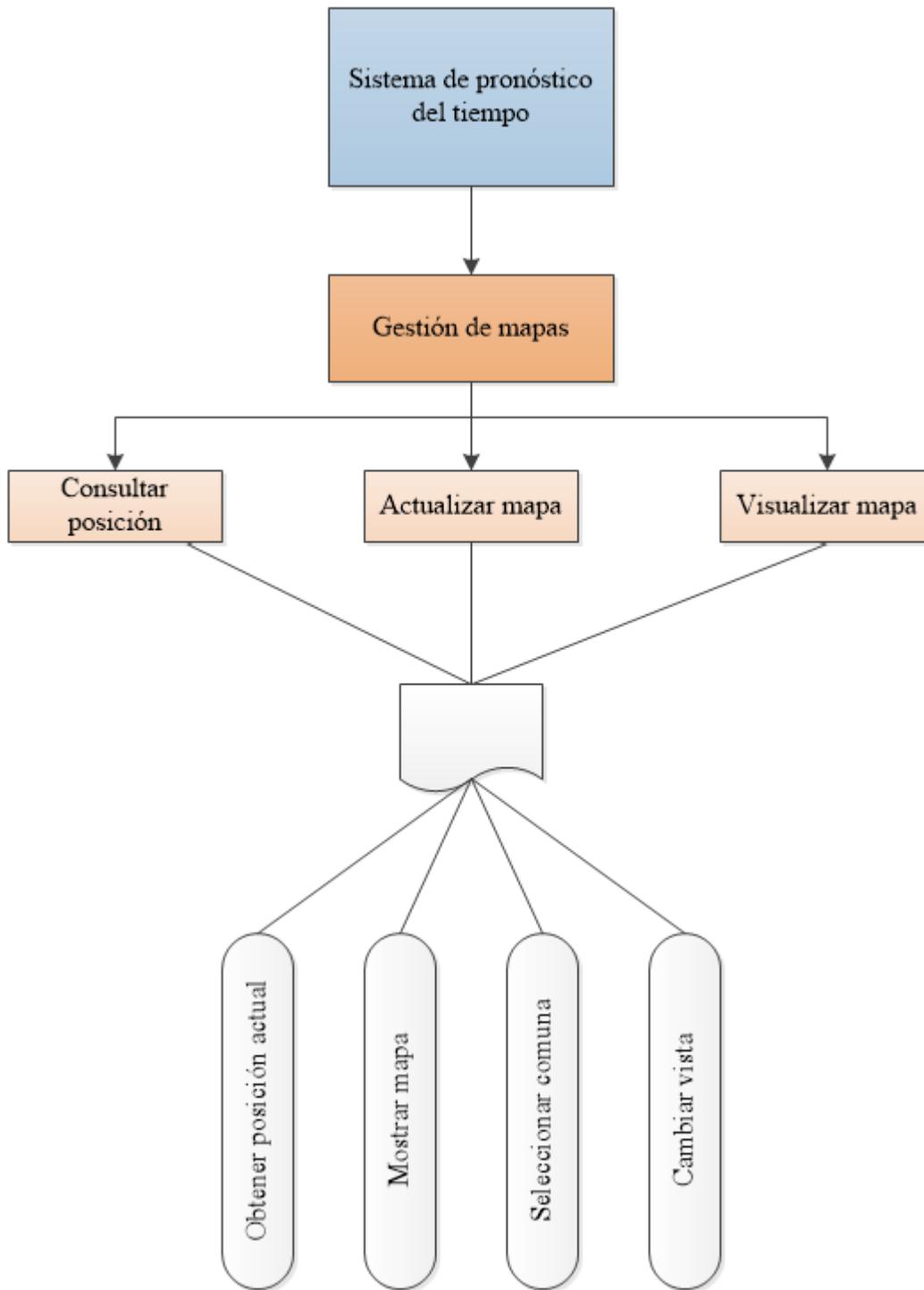


Figura 7.4: Descomposición funcional - Gestión de mapas.

**Fuente:** Elaboración propia.

## 7.3. Diseño de interfaz y navegación

En este apartado se da a conocer la interfaz que presenta la aplicación móvil considerando la ubicación de los respectivos elementos a visualizar y la iconografía implementada.

Cabe destacar que las imágenes que se presentan a continuación están sujetas a posibles cambios durante el proceso de desarrollo.

### 7.3.1. iconografía

Logo de la aplicación móvil.

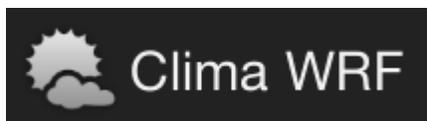


Figura 7.5: Logo de la aplicación.

**Fuente:** elaboración propia.

Posibles figuras a utilizar en la visualización del pronóstico del tiempo.



Figura 7.6: íconos para representar pronóstico.

**Fuente:** Página web [15].

Otras figuras posiblemente utilizadas en la interfaz y visualización del pronóstico.



Figura 7.7: figura de temperatura.



Figura 7.8: ícono de mapa.



Figura 7.9: ícono de ubicación.



Figura 7.10: ícono de menú.

Figura 7.11: Otras figuras.

**Fuente:** Android Design.

La interfaz de la aplicación se explica a continuación.

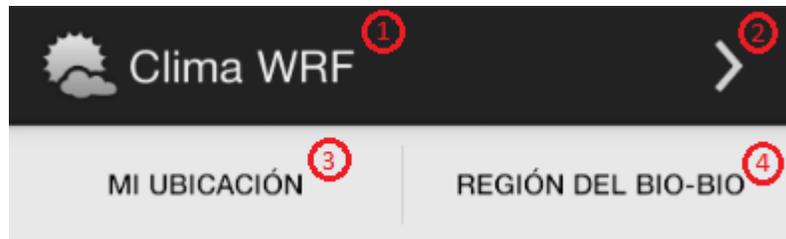


Figura 7.12: Barra de la aplicación android.

**Fuente:** Elaboración propia.

1. Logo de la aplicación.
2. Ícono para desplegar menú de solicitud de pronóstico.
3. Pestaña que muestra la información de la localidad donde se encuentra el usuario.
4. Pestaña que muestra el mapa de la región del Bío-Bío, mediante la API de google maps.

El mapa puede cambiar su forma de visualizar mediante la opción de “cambiar vista”. En la figura 7.13 se presenta las opciones de vista con las que se cuenta.



Figura 7.13: tipos de vistas del mapa.

**Fuente:** Elaboración propia.

en cuanto al despliegue del pronóstico se encuentran dos formas de representar la información:

- Para el pronóstico de un día, la información es mayor por lo que cada ítem de la lista se verá con el formato que sigue a continuación.



Figura 7.14: Formato para pronóstico de un día.

**Fuente:** Elaboración propia.

- Para el pronóstico de 3 días, la lista cuenta con solo tres elementos (uno por cada día) por lo que se utiliza un poco más de espacio en la plantilla para desplegar cada ítem de la lista.



Figura 7.15: Formato para pronóstico de más días.

**Fuente:** Elaboración propia.

donde la imagen (figura Android) será reemplazada con la iconografía correspondiente al pronóstico climático (imagen del tipo de la figura 7.6).

## 7.4. Especificación de módulos

Nombre módulo: Extraer datos			
Parametros de entrada		Parametros de salida	
Nombre:	Tipo de datos:	Nombre:	Tipo de datos:
comuna	String	"nombreComuna".txt	Archivo
latitud	Double		
longitud	Double		

Tabla 7.1: Especificación de módulos - Extraer datos.

**Fuente:** Elaboración propia.

Nombre módulo: Realizar solicitud			
Parametros de entrada		Parametros de salida	
Nombre:	Tipo de datos:	Nombre:	Tipo de datos:
comuna	String	resultado	Boolean

Tabla 7.2: Especificación de módulos - Realizar solicitud.

**Fuente:** Elaboración propia.

Nombre módulo: Calcular datos			
Parametros de entrada		Parametros de salida	
Nombre:	Tipo de datos:	Nombre:	Tipo de datos:
comuna	String	datos	Array

Tabla 7.3: Especificación de módulos - Calcular datos.

**Fuente:** Elaboración propia.

Nombre módulo: Recibir datos			
Parametros de entrada		Parametros de salida	
Nombre:	Tipo de datos:	Nombre:	Tipo de datos:
		resultado_consulta	Array

Tabla 7.4: Especificación de módulos - Recibir datos.

**Fuente:** Elaboración propia.

Nombre módulo: Desplegar información			
Parametros de entrada		Parametros de salida	
Nombre:	Tipo de datos:	Nombre:	Tipo de datos:
resultado_consulta	Array	imagen	Drawable
		hora	String
		temperatura	String
		humedad	String
		presion	String
		velocidad	String
		direccion	String

Tabla 7.5: Especificación de módulos - Desplegar información.

**Fuente:** Elaboración propia.

Nombre módulo: Obtener posición actual			
Parametros de entrada		Parametros de salida	
Nombre:	Tipo de datos:	Nombre:	Tipo de datos:
		latitud	Double
		longitud	Double

Tabla 7.6: Especificación de módulos - Obtener posición actual.

**Fuente:** Elaboración propia.

Nombre módulo: Mostrar mapa			
Parametros de entrada		Parametros de salida	
Nombre:	Tipo de datos:	Nombre:	Tipo de datos:
		respuesta	Boolean

Tabla 7.7: Especificación de módulos - Mostrar mapa.

**Fuente:** Elaboración propia.

Nombre módulo: Seleccionar comuna			
Parametros de entrada		Parametros de salida	
Nombre:	Tipo de datos:	Nombre:	Tipo de datos:
comuna	String	respuesta	Boolean

Tabla 7.8: Especificación de módulos - Seleccionar comuna.

**Fuente:** Elaboración propia.

Nombre módulo: Cambiar vista			
Parametros de entrada		Parametros de salida	
Nombre:	Tipo de datos:	Nombre:	Tipo de datos:
		respuesta	Boolean

Tabla 7.9: Especificación de módulos - Cambiar vista.

**Fuente:** Elaboración propia.

## Capítulo 8

# Pruebas

Adaptación basada en *IEEE Software Test Documentation Std 829-1998*.

### 8.1. Introducción

A continuación se especifican los elementos de prueba, su detalle, las personas responsables de ello y la calendarización establecida.

### 8.2. Elementos de pruebas

1. **Nombre:** Conexión con Servidor.

**Descripción:** Se comprueba el correcto funcionamiento respecto a la conectividad del dispositivo con el servidor.

2. **Nombre:** Interfaz.

**Descripción:** Todos los elementos deben ser visualizados correctamente.

3. **Nombre:** Información de pronóstico.

**Descripción:** El calculo para la información del pronóstico debe ser correcto.

4. **Nombre:** Rutina Matlab.

**Descripción:** La rutina Matlab debe funcionar correctamente.

### 8.3. Especificación de pruebas

ID	Características a probar	Nivel de prueba	Objetivo de la prueba	Enfoque para la definición de casos de prueba
1	Funcionalidad	Unidad	Comprobar si se establece conexión con el servidor.	Caja negra
2	Interfaz	Unidad	Detectar fallas de tipo estético y/o visualización de la información.	Caja negra
3	Funcionalidad	Unidad	Comprobar el correcto calculo de los datos para el pronóstico.	Caja negra
4	Funcionalidad	Unidad	Detectas posibles errores al crear los archivos con Matlab.	Caja negra

Tabla 8.1: Especificación de las pruebas - parte 1.

**Fuente:** Elaboración propia.

ID	Técnicas para la definición de casos de prueba	Actividades de prueba	Criterios de cumplimiento
1	Valores límites y particiones	1.- Probar en ambas pestañas de la aplicación si se establece la correcta conexión con el servidor, para los dos tipos de pronósticos (uno o más días).	Se debe establecer conexión con el servidor.
2	Valores límites y particiones	1.- Interacción con el mapa de Google maps, que los elementos de desplieguen de forma correcta. 2.- La correcta visualización de la información del pronóstico del tiempo.	clara visualización de la información del pronóstico.
3	Valores límites y particiones	1.- Según las variables de viento, comprobar si el calculo de la dirección del viento es correcta. 2.- Comprobar el correcto calculo de las temperaturas mínimas y máximas, o promedio según corresponda. 3.- Validar el calculo de la humedad.	La información reunida debe ser acorde a los datos utilizados.
4	Valores límites y particiones	1.- Verificar la creación de archivos a partir de los datos guardados en los archivos NetCDF.	Todos los archivos deben ser creados correctamente.

Tabla 8.2: Especificación de las pruebas - parte 2.

Fuente: Elaboración propia.

## 8.4. Responsables de las pruebas

Referente a las pruebas establecidas en la tabla 3 se presenta los responsables por bloque de pruebas.

ID	Responsable
1	Paola Angélica Torres Ferrada
2	Paola Angélica Torres Ferrada
3	Renato Joaquín Ávila Momberg
4	Renato Joaquín Ávila Momberg

Tabla 8.3: Responsables de las pruebas a realizar.

**Fuente:** Elaboración propia.

## 8.5. Calendario de pruebas

ID	Fecha
1	12 de Febrero de 2014
2	12 de Febrero de 2014
3	10 de Febrero de 2014
4	10 de Febrero de 2014

Tabla 8.4: Calendario de las pruebas a realizar.

**Fuente:** Elaboración propia.

## 8.6. Detalle de pruebas

Este apartado se especifica en el anexo 3.

## 8.7. Conclusión de pruebas

Las pruebas son una parte fundamental en cualquier desarrollo de software aunque muchas veces queda relegada por priorizar otros elementos.

En este caso las pruebas han resultado satisfactorias en su totalidad, lo que verifica el correcto funcionamiento del software y sus partes.

De todas formas se continúa monitoreando ante cualquier tipo de error inesperado en el software.

## Capítulo 9

# Plan de Implementación y puesta en marcha

Los pasos que se deben seguir para una correcta implementación y puesta en marcha son los siguientes:

1. Agregar a la rutina existente encargada de interactuar con el modelo WRF los archivos matlab para extraer los datos resultantes del modelo en los archivos de texto respectivos.
2. Subir al servidor del departamento de Geofísica de la Universidad de Concepción la pagina web encargada de la interacción entre aplicación móvil y datos extraídos del modelo.
3. Publicar la aplicación móvil en la tienda “Google Play” siguiendo los pasos impuestos por la tienda entre los cuales destaca el pago de la suscripción por parte de la cuenta de correos de Google (previamente estipulado en el estudio de factibilidad).
4. Comprobar el correcto funcionamiento de la aplicación previamente publicada.

## Capítulo 10

# Resumen de esfuerzo requerido

En el presente capítulo se dan a conocer las horas de trabajo destinadas a la realización del proyecto, tanto del grupo en total, como también por cada integrante.

- Nombre integrante: Paola Torres Ferrada.

Actividades/fases	Nº Horas
Reuniones con cliente	15
Reuniones de grupo	140
Investigación sobre metodología	23
Diseño y modelado de interface	100
Preparación informe	150
Programación y estudio de lenguajes	580
Investigación previa del tema y estado del arte	50
<b>Total</b>	<b>1058</b>

Tabla 10.1: Resumen de esfuerzo requerido de Paola Torres Ferrada.

**Fuente:** Elaboración propia.

- Nombre integrante: Renato Ávila Momberg.

Actividades/fases	N° Horas
Reuniones con cliente	15
Reuniones de grupo	140
Investigación sobre metodología	17
Diseño y modelado de interface	100
Preparación informe	150
Programación y estudio de lenguajes	620
Investigación previa del tema y estado del arte	40
<b>Total</b>	<b>1082</b>

Tabla 10.2: Resumen de esfuerzo requerido de Renato Ávila Momberg.

**Fuente:** Elaboración propia.

- Horas totales.

Actividades/fases	N° Horas
Reuniones con cliente	30
Reuniones de grupo	280
Investigación sobre metodología	40
Diseño y modelado de interface	200
Preparación informe	300
Programación y estudio de lenguajes	1200
Investigación previa del tema y estado del arte	90
<b>Total</b>	<b>2140</b>

Tabla 10.3: Total de esfuerzo requerido.

**Fuente:** Elaboración propia.

## Capítulo 11

# Conclusiones

Basándose en la problemática planteada por el departamento de Geofísica de la Universidad de Concepción, los objetivos del proyecto y del software, el producto desarrollado cumple con todo lo planteado en los inicios entregando al usuario la información correspondiente al pronóstico del tiempo de forma clara y precisa, con una interfaz gráfica simple e intuitiva para quien haga uso de la aplicación.

Respecto al trabajo realizado, se ha aplicado lo aprendido y estudiado durante estos cinco años dentro de la carrera de Ingeniería Civil en Informática, y a pesar de que en un principio del proyecto el equipo de trabajo no dominaba los lenguajes y herramientas utilizadas, la experiencia adquirida con los años en la Universidad del Bío-Bío motiva al equipo a aceptar el reto y ser capaces de finalizar con éxito esta gran etapa.

El proyecto de título ha sido un trabajo arduo debido a los variados componentes que se tuvieron que manejar a lo largo del desarrollo del proyecto dentro los cuales destacan la realización de una encuesta a potenciales usuarios con la cual se pudo respaldar los requerimientos y poder conocer los hábitos y puntos de vista respecto de los pronósticos del tiempo, también en cuanto a la programación se realizó una ardua investigación para la realización de la aplicación y manejo de los componentes involucrados como el uso de una arquitectura de software, implementación de patrones de diseño, etc. Por otra parte el tener que introducirse en un área que es completamente distinta, como lo es la meteorología resulta ser un gran crecimiento en cuanto a lo profesional y así ampliar los horizontes en cuanto al conocimiento.

El proyecto no fue fácil de desarrollar, no solo por el hecho de introducirse en el área de la

---

meteorología en términos de teoría, sino que por el funcionamiento interno del modelo de predicción utilizado. El problema de la extracción de datos a partir de los archivos NetCDF generó un gran obstáculo. Pero al seguir un plan de trabajo basado en la metodología ágil de “programación extrema” el equipo continuó avanzando en el proyecto sin quedarse estancado, buscando la solución al problema surgido, pero por el hecho de las constantes reuniones con el profesor co-guía y a medida que el equipo desarrollador comprendía el modelo y su funcionamiento se pudo superar con éxito el obstáculo de la extracción de datos y continuar avanzando de forma fluida para dar un buen término al proyecto.

En términos personales haber finalizado no solo el proyecto de título, sino que todo un ciclo de continuo aprendizaje en la Universidad del Bío-Bío se convierte en una experiencia tremendamente positiva y valorada por quienes participamos de este proyecto de título.

## Referencias

1. Universidad de Concepción. Plan estratégico. [en línea] <[http://www.udec.cl/intranet/documentos\\_oficiales/PEI\\_2011\\_2015\\_web.pdf](http://www.udec.cl/intranet/documentos_oficiales/PEI_2011_2015_web.pdf)> [Consulta: 16 Diciembre 2013]
2. Universidad de Concepción. Organigrama de la corporación [en línea] <<http://www.udec.cl/pexterno/node/3?q=node/144>> [Consulta: 01 Septiembre 2013]
3. Universidad de Concepción. Estructura orgánica de la Universidad de Concepción. [en línea] <<http://www.udec.cl/pexterno/node/1?q=node/132>> [Consulta: 01 Septiembre 2013]
4. Departamento de GeoFísica. Historia del departamento. [en línea] <<http://www.dgeo.udec.cl/nosotros/historia/>> [Consulta: 05 Septiembre 2013]
5. Departamento de GeoFísica. Regional Ocean and Atmosphere Numerical Modeling(ROIN) [en línea] <<http://ronin.dgeo.udec.cl:8080/dgeo/#!webContent;id=2>> [Consulta: 01 Octubre 2013]
6. University Corporation for Atmospheric Research (UCAR). NetCDF. [en línea] <<http://www.unidata.ucar.edu/software/netcdf/>> [Consulta: 01 Octubre 2013]
7. Windows. Características de Windows Phone (Chile). [en línea] <<http://www.windowsphone.com/es-cl/features>> [Consulta: 26 Enero 2014]
8. Apple. iPhone. [en línea] <<http://www.apple.com/iphone/>> [Consulta: 26 Enero 2014]
9. Android. Frequently Asked Questions — Android Developers. [en línea] <<http://source.android.com/source/faqs.html>> [Consulta: 26 Enero 2014]
10. BlackBerry. BlackBerry Chile - Smartphones BlackBerry Z10 - Modelos de Celulares - Chile. [en línea] <<http://cl.blackberry.com/>> [Consulta: 26 Enero 2014]
11. Android. Android SDK. [en línea] <<http://developer.android.com/sdk/index.html#ExistingIDE>> [Consulta: 28 Enero 2014]

12. Android. Using the Emulator. [en línea] <<http://developer.android.com/tools/devices/emulator.html#accel-graphics>> [Consulta: 28 Enero 2014]
13. WELLS, Don. Extreme Programming. [en línea] <<http://www.extremeprogramming.org/>> [Consulta: 05 Octubre 2013]
14. IDC Worldwide Mobile Phone Tracker. [en línea] <<http://www.idc.com/getdoc.jsp?containerId=prUS24257413>> [Consulta: 20 Septiembre 2013]
15. MerlinTheRed. plain weather icons. [en línea] <<http://merlinthered.deviantart.com/art/plain-weather-icons-157162192>> [Consulta: 27 Septiembre 2013]
16. Pressman, Roger S. Ingeniería del software: un enfoque práctico. Quinta edición. México: McGraw-Hill, 2010. 777 p. Título original: Software engineering. A practitioner's approach.
17. Ramos, Daniel. y Muñoz, Boris. Sistema de apoyo a la administración y búsqueda de estacionamientos privados en la ciudad de Concepción con utilización de dispositivos móviles. Tesis (Ingeniería Civil en Informática). Concepción: Universidad del Bío-Bío. Depto. de Sistemas de Información, 2013.
18. Martínez, Paola. y Valenzuela, Fabián. Sistema de apoyo a la primera respuesta ante incidentes con materiales peligrosos. Tesis (Ingeniería Civil en Informática). Concepción: Universidad del Bío-Bío. Depto. de Sistemas de Información, 2013.
19. A Description of the Advanced Research WRF Version 3.2008. Por Skamarock, William C. "et al". National Center for Atmospheric Research, Boulder, Colorado, USA. 125p.
20. MATLAB - El lenguaje de cálculo técnico - MathWorks España. [en línea] <<http://www.mathworks.es/products/matlab/>> [Consulta: 01 Octubre 2013]
21. Google Play. [en línea] <<https://play.google.com/store>> [Consulta: 10 Septiembre 2013]
22. Unified Modeling Language (UML). [en línea] <<http://www.uml.org/>> [Consulta: 27 Septiembre 2013]
23. IDC Says. [en línea] <<http://www.idc.com/getdoc.jsp?containerId=prUS23916413>> [Consulta: 11 Septiembre 2013]
24. IDC Home: The premier global market intelligence firm. [en línea] <<http://www.idc.com/>> [Consulta: 9 Septiembre 2013]

25. The Weather Research&Forecasting Model Website. [en línea] <<http://www.wrf-model.org/index.php>> [Consulta: 1 Septiembre 2013]
26. ISO - International Organization for Standardization. [en línea] <<http://www.iso.org/iso/home.html>> [Consulta: 16 Diciembre 2013]
27. IEEE - The world's largest professional association for the advancement of technology. [en línea] <<http://www.ieee.org/index.html>> [Consulta: 16 Diciembre 2013]
28. Descarga gratuita de software de Java. [en línea] <<https://www.java.com/es/download/>> [Consulta: 16 Diciembre 2013]
29. PHP: Hypertext Preprocessor. [en línea] <<http://www.php.net/>> [Consulta: 16 Diciembre 2013]
30. Java SE - Downloads — Oracle Technology Network — Oracle. [en línea] <<http://www.oracle.com/technetwork/es/java/javase/downloads/index.html>> [Consulta: 16 Diciembre 2013]
31. XML.com. [en línea] <<http://www.xml.com/pub/a/98/10/guide0.html>> [Consulta: 16 Diciembre 2013]
32. Near Field Communication: What is Near Field Communication?. [en línea] <<http://www.nearfieldcommunication.org/>> [Consulta: 16 Diciembre 2013]
33. Apple - Mac. [en línea] <<https://www.apple.com/mac/>> [Consulta: 16 Diciembre 2013]
34. Welcome to The Apache Software Foundation!. [en línea] <<http://www.apache.org/>> [Consulta: 16 Diciembre 2013]
35. API de Google Maps — Google Developers. [en línea] <<https://developers.google.com/maps/?hl=es>> [Consulta: 16 Diciembre 2013]
36. Definicion de IDE. [en línea] <<http://www.alegsa.com.ar/Dic/ide.php>> [Consulta: 15 Noviembre 2013]
37. Definicion de GUI. [en línea] <<http://www.alegsa.com.ar/Dic/gui.php>> [Consulta: 15 Noviembre 2013]
38. Definicion de GPS. [en línea] <<http://www.alegsa.com.ar/Dic/gps.php>> [Consulta: 15 Noviembre 2013]

39. gnu.org. [en línea] <<http://www.gnu.org/home.es.html>> [Consulta: 16 Diciembre 2013]
40. Definicion de Smarthphone. [en línea] <<http://www.alegsa.com.ar/Dic/smarthphone.php>> [Consulta: 15 Noviembre 2013]

## Anexo 1

# Planificación inicial del proyecto

	Nombre de tarea	Duración	Comienzo	Fin	
Diagrama de Gantt	1	Entrega Propuesta	1 día	lun 26-08-13	lun 26-08-13
	2	Resolución Propuesta	1 día	vie 30-08-13	vie 30-08-13
	3	Estudio con Profesor Co-guía del modelo de analisis climático y al extracción de datos	45 días	lun 02-09-13	vie 01-11-13
	4	Recopilación y documentación de información de Lenguaje de programación en Android	19 días	mar 27-08-13	vie 20-09-13
	5	clasificar y estudiar información recopilada sobre programación	20 días	lun 02-09-13	vie 27-09-13
	6	Especificación de requerimientos	15 días	lun 02-09-13	vie 20-09-13
	7	Estudio de Factibilidades	15 días	lun 16-09-13	vie 04-10-13
	8	Análisis (realización de diagramas)	15 días	lun 30-09-13	vie 18-10-13
	9	Diseño	15 días	lun 14-10-13	vie 01-11-13
	10	Programación en Android	50 días	lun 30-09-13	vie 06-12-13
	11	Pruebas	15 días	lun 25-11-13	vie 13-12-13
	12	Revisión, plan de implantación y puesta en marcha.	15 días	lun 02-12-13	vie 20-12-13
	13	Estructuración final de informe	5 días	mié 18-12-13	mar 24-12-13
	14	Entrega Informe de Tesis	1 día	vie 27-12-13	vie 27-12-13

Tabla 1.1: Planificación inicial.

**Fuente:** Elaboración propia.

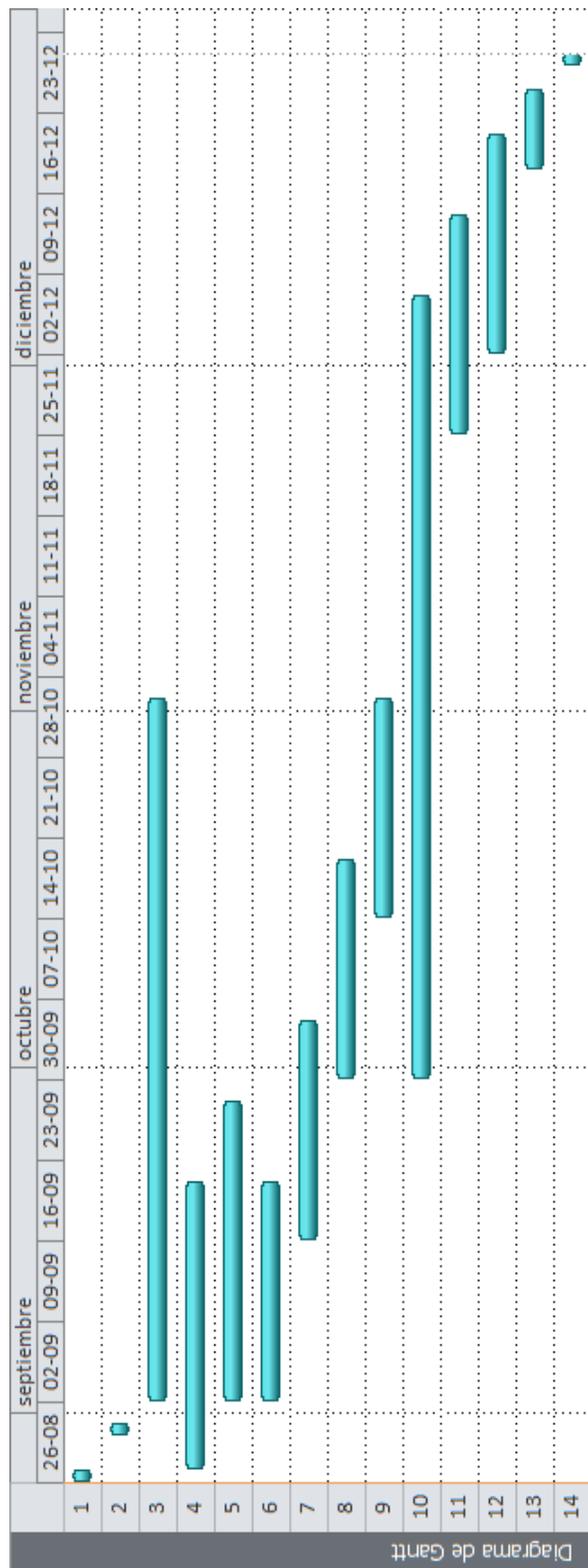


Figura 1.1: Planificación inicial.

Fuente: Elaboración propia.

## 1.1. Estimación inicial de tamaño

Para medir el tamaño del sistema, se utiliza la estimación de Puntos de Caso de Uso, los cuales serán explicados a continuación:

### 1.1.1. Clasificación de los actores

Actor	Nivel de complejidad	Factor
Usuario	Complejo	3
Modelo WRF	Complejo	3
Google Maps	Simple	1
Sensor GPS	Medio	2

Tabla 1.2: Clasificación de los actores.

En base a la tabla 1.2 se puede calcular el factor de peso de los actores sin ajustar (UAW), mediante la suma de valores asignados:

$$\mathbf{UAW} = 3 + 3 + 1 + 2 = 9$$

### 1.1.2. Clasificación de los casos de uso

Caso de uso	Tipo de caso de uso	Factor
Datos Modelo	Simple	5
Calcular Información	Simple	5
Gestión Solicitud	Medio	10
Consultar Posición	Simple	5
Consultar Pronostico	Simple	5
Mostrar Información	Simple	5
Seleccionar Comuna	Simple	5
Visualizar Mapa	Simple	5
Actualizar Mapa	Simple	5
Cambiar Vista Del Mapa	Simple	5
Consultar Instrucciones	Simple	5

Tabla 1.3: Clasificación de los casos de uso.

El factor de peso de los casos de uso sin ajustar (UUCW) se calcula también por la suma de factores asignados, resultando:

$$\text{UUCW} = (10 * 5) + 10 = 60$$

### 1.1.3. Puntos de caso de uso sin ajustar

Los puntos de casos de uso sin ajustar se calculan mediante la siguiente fórmula:

$$\text{UUCP} = \text{UAW} + \text{UUCW}$$

Entonces

$$\text{UUCP} = 9 + 60 = 69$$

#### 1.1.4. Factores técnicos

Los factores técnicos tienen relevancia con la complejidad del software a construir, y según sean los requerimientos que se planteen. El factor de complejidad técnica (TCF) está determinado por:

$$\mathbf{TCF} = 0,6 + (0,01 * TFactor)$$

Donde *TFactor* es:

$$\mathbf{TFactor} = \sum (Multiplicador * Factor)$$

Para la evaluación de los factores técnicos y ambientales se utilizará la siguiente escala de valores:

Descripción	Valor
Irrelevante	De 0 a 2
Medio	De 3 a 4
Esencial	5

Tabla 1.4: Escala de evaluación para factores técnicos y ambientales.

En la tabla 1.5 se presenta cada ítem de los factores técnicos con su respectivo factor asociado.

Id	Descripción	Multiplicador	Factor
T1	Sistema distribuido	2	1
T2	Objetivos de desempeño o tiempo de respuesta	1	3
T3	Eficiencia del usuario final	1	5
T4	Procesamiento interno complejo	1	3
T5	El código debe ser reutilizable	1	2
T6	Facilidad de instalación	0.5	3
T7	Facilidad de uso	0.5	5
T8	Portabilidad	2	3
T9	Facilidad de cambio	1	3
T10	Concurrencia	1	5
T11	Objetivos especiales de seguridad	1	1
T12	Provee acceso directo a terceras partes	1	1
T13	Se requiere facilidades especiales de entrenamiento a usuario	1	1

Tabla 1.5: Factores técnicos.

Entonces  $TFactor$  es igual a 36.

$$TCF = 0,6 + (0,01 * 36) = 0,96$$

### 1.1.5. Factores ambientales

El factor ambiental (EF) está determinado por:

$$EF = 1,4 + (-0,03 * EFactor)$$

Donde  $EFactor$  es:

$$EFactor = \sum (Multiplicador * Factor)$$

Id	Descripción	Multiplicador	Factor
E1	Familiaridad con el modelo de proyecto utilizado	1.5	2
E2	Experiencia en la aplicación	0.5	2
E3	Experiencia en orientación a objetos	1	4
E4	Capacidad del analista líder	0.5	4
E5	Motivación	1	5
E6	Estabilidad de los requerimientos	2	4
E7	Personal part-time	-1	3
E8	Dificultad del lenguaje de programación	-1	4

Tabla 1.6: Factores ambientales.

Entonces  $E_{Factor}$  es igual a 16.

$$EF = 1,4(-0,03 * 16) = 0,92$$

#### 1.1.6. Puntos de caso de uso ajustados

$$UCP = UUCP * EF * TCF$$

$$UCP = 69 * 0,92 * 0,96$$

$$UCP = 60,9408$$

#### 1.1.7. Cálculo de esfuerzo requerido

Se considera contar el factor según los siguientes criterios:

- Contar los factores desde E1 hasta E6 que sean menor a 3.
- Contar los factores desde E7 hasta E8 que sean mayores a 3.

Los factores de la tabla 1.6 que cumplen con los criterios mencionados son los factores E1, E2 y E8.

LOE	Filtro
20	De 0 a 2
28	De 3 a 4
Reconsiderar	Mayor a 4

Tabla 1.7: Valores de LOE.

Considerando la tabla 1.7 y teniendo presente que son 3 los elementos que cumplen los criterios, se establece un valor de LOE igual a 28.

Por lo que el total de horas es:

$$\mathbf{horas} = UCP * LOE$$

$$\mathbf{horas} = 60,9408 * 28$$

$$\mathbf{horas} = 1706,34$$

#### 1.1.8. Contabilización final del tamaño del software

El proyecto cuenta con 7658 líneas de código, esto es considerando líneas de comentarios y las que son en blanco con el fin de separar las funciones y métodos. El conteo de líneas considera todos los lenguajes utilizados que son Android, PHP y Matlab.

## Anexo 2

# Encuesta a potenciales usuarios

### 2.1. Introducción

Para validar y/o complementar los requerimientos y apreciar la opinión de potenciales usuarios de la aplicación se realiza una encuesta de tipo online.

### 2.2. Datos de la encuesta

- **Realizada por:** Paola Torres - Renato Ávila.
- **Universo:** Mercado potencial (usuarios de la aplicación móvil).
- **Unidad de muestreo:** Cantidad de personas.
- **Fecha:** desde el 17 de Septiembre de 2013 hasta el 5 de Octubre de 2013.
- **Tipo de muestreo:** No probabilístico.
- **Técnica de recolección de datos:** Encuesta online.
- **Tamaño de la muestra:** 104 personas.
- **Objetivos de la encuesta:**
  1. Evaluar el interés por la aplicación propuesta.
  2. Complementar/Validar requerimientos para el desarrollo de la aplicación móvil.
- **Nº de preguntas formuladas:** 10 preguntas.

### 2.3. Análisis de respuestas

■ **Pregunta 1:**

**¿Cuál es su rango de edad?**

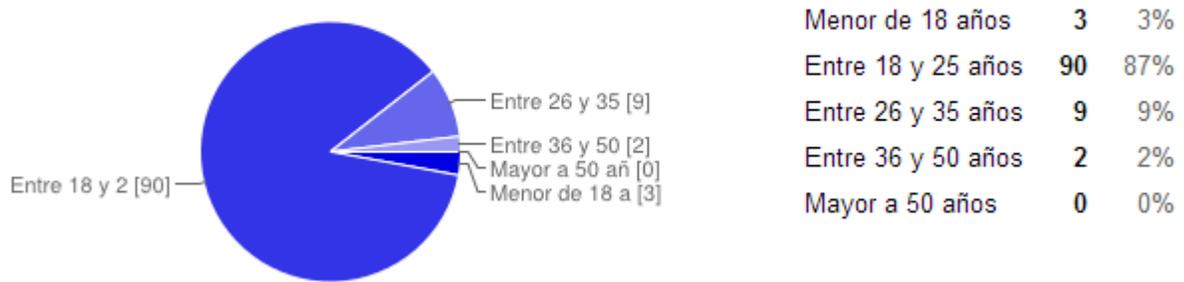


Figura 2.1: Pregunta 1.

Analizando la figura 2.1 se puede notar lo siguiente:

- Aproximadamente el 97% de las personas encuestadas son mayores de edad, dentro de los cuales prácticamente su totalidad comprenden el rango de edad de entre los 18 y 25 años, sección de público que se puede asumir como altamente tecnologizado.

■ **Pregunta 2:**

**¿Posee usted algún dispositivo con Sistema Operativo Android?**

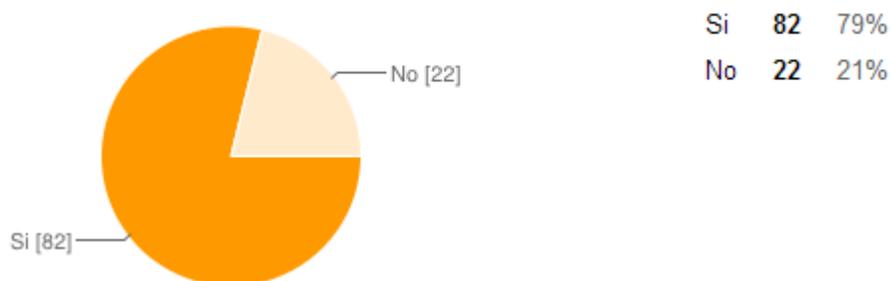


Figura 2.2: Pregunta 2.

Analizando la figura 2.2 se puede notar lo siguiente:

- En su mayoría, quienes contestaron la encuesta poseen algún dispositivo con Sistema Operativo Android, lo que los convierte inmediatamente en potenciales usuarios de la

aplicación propuesta en este proyecto.

- Respuestas de poco más de 100 personas pueden informar de una tendencia, lo que se comprueba con la información recopilada del uso de la plataforma Android a nivel mundial en el último tiempo.

▪ **Pregunta 3:**

**¿Que versión utiliza su dispositivo Android?**

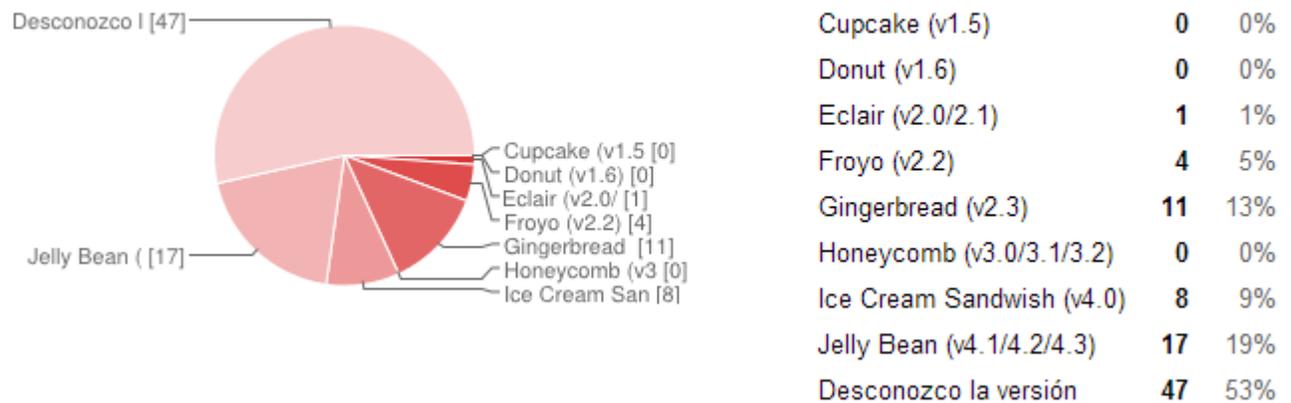


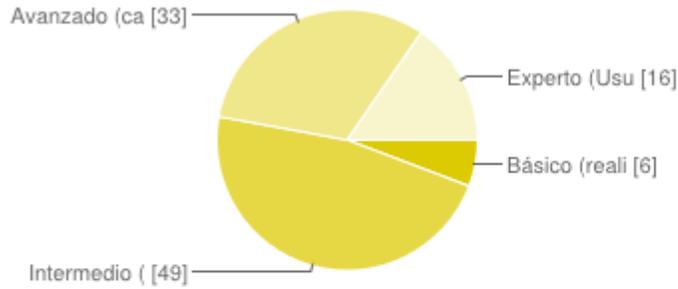
Figura 2.3: Pregunta 3.

Analizando la figura 2.3 se puede notar lo siguiente:

- Un alto porcentaje de encuestados desconoce la versión de su dispositivo Android, pero considerando las respuestas que sí marcan una versión, la mayoría se inclina por “Jelly Bean”, comparando estos resultados con los que posee el sitio oficial de Android, se corrobora que la versión más utilizada es efectivamente “Jelly Bean”.
- Muchas son las personas encuestadas que teniendo un dispositivo con sistema operativo Android desconoce cual es la versión del mismo.

▪ **Pregunta 4:**

**Según Ud. ¿cuál es su nivel de usuario?**



Básico (realizar llamadas, enviar mensajes de texto, agregar contactos)	6	6%
Intermedio (descargar aplicaciones, activar funcionalidades, navegar en internet)	49	47%
Avanzado (cambiar configuración, respaldar información, traspaso de archivos)	33	32%
Experto (Usuario root, instalar ROMS)	16	15%

Figura 2.4: Pregunta 4.

Analizando la figura 2.4 se puede notar lo siguiente:

- Considerando que más del 90% posee un nivel de usuario Intermedio o superior, se puede interpretar a estas personas como usuarios habituales de dispositivos inteligentes por lo que no tendrían ningún inconveniente en adaptarse a la aplicación propuesta en este proyecto.
- En cuanto a los usuarios básicos que posean un smartphone y tablet con Android, tampoco deberían presentar algún tipo de impedimento para utilizar la aplicación porque está pensada para ser muy intuitiva.

■ **Pregunta 5:**

**¿Con cuanta frecuencia revisa pronósticos del tiempo?**

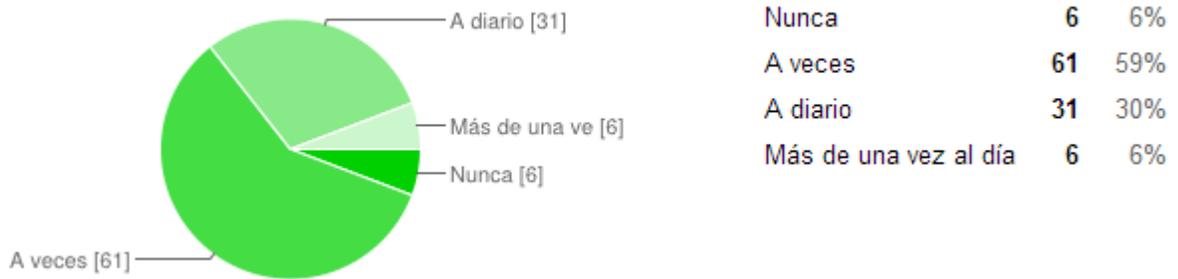


Figura 2.5: Pregunta 5.

Analizando la figura 2.5 se puede notar lo siguiente:

- La mayoría de las personas encuestadas en algún momento consultan pronósticos del tiempo.

■ **Pregunta 6:**

**¿Donde revisa el pronóstico?**

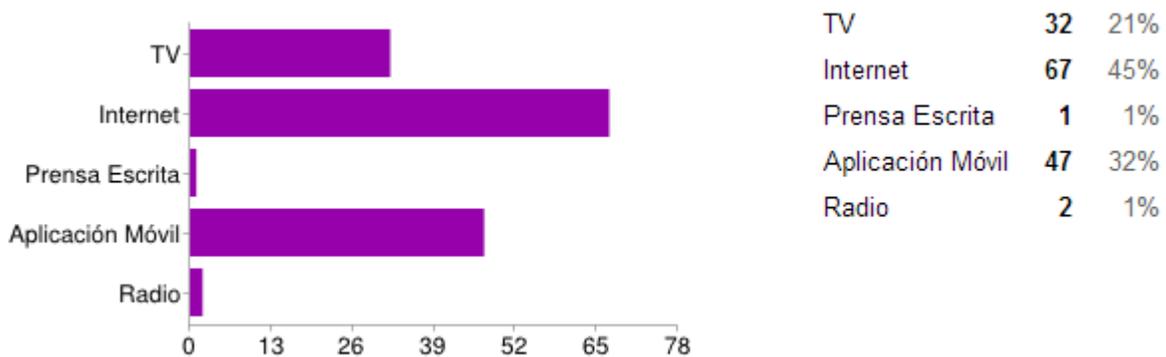


Figura 2.6: Pregunta 6.

Analizando la figura 2.6 se puede notar lo siguiente:

- La concentración de opciones está en internet y aplicaciones móviles, en menor medida se considera a la televisión para revisar el pronóstico del tiempo y prácticamente nula

la participación de la prensa escrita y la radio. Esto demuestra el auge de la tecnología y la dependencia de las personas hacia la tecnología.

▪ **Pregunta 7:**

**¿Cuanto confía en el pronóstico?**

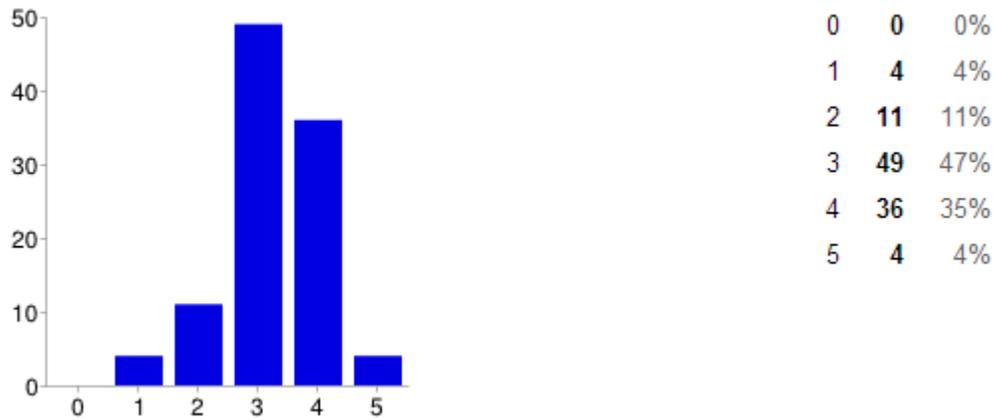


Figura 2.7: Pregunta 7.

Analizando la figura 2.7 se puede notar lo siguiente:

- Hay muchas personas que no tienen plena confianza en los pronósticos que revisan.

▪ **Pregunta 8:**

**¿Posee alguna aplicación en su teléfono para consultar el tiempo?**

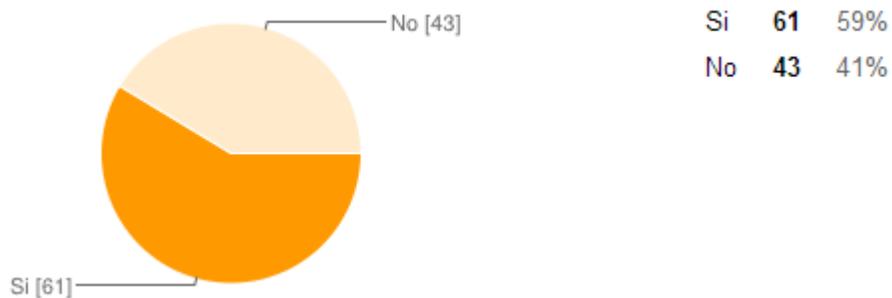


Figura 2.8: Pregunta 8.

Analizando la figura 2.8 se puede notar lo siguiente:

- Un alto porcentaje utiliza una aplicación para consultar el pronóstico, pero no se puede dejar pasar el 43% que no utiliza, es por esto que con la aplicación buscará se incrementa el número de personas que utiliza una aplicación para el clima.

▪ **Pregunta 9:**

**¿La aplicación es gratuita, o de pago?**

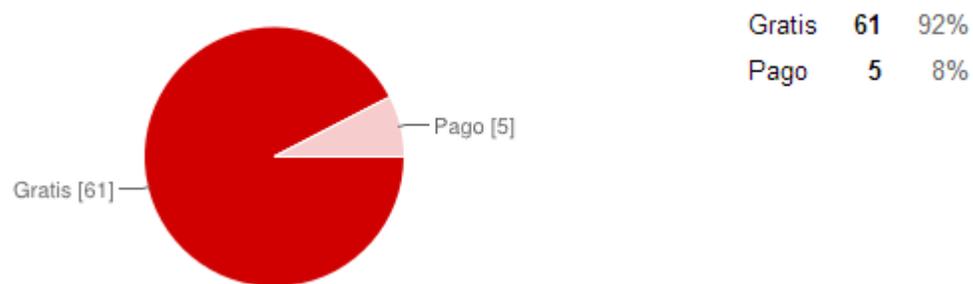


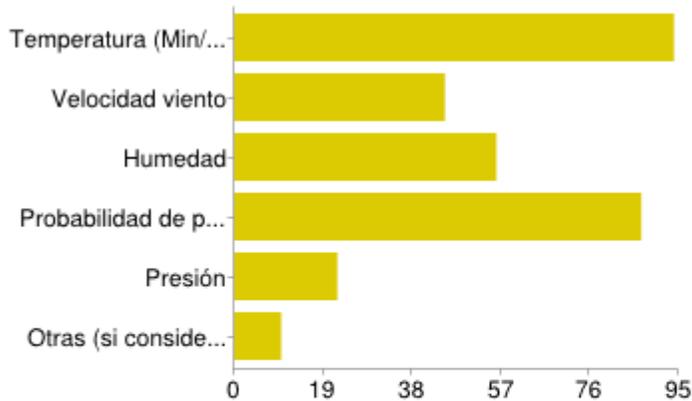
Figura 2.9: Pregunta 9.

Analizando la figura 2.9 se puede notar lo siguiente:

- Existen personas que están dispuestas a pagar por una aplicación de este tipo, pero es más usual utilizar una gratuita ya que siempre se está buscando en estos casos una aplicación de calidad pero que a su vez no suponga un gasto para el usuario.

▪ **Pregunta 10:**

**¿Que característica debería presentar la aplicación?**



Temperatura (Min/Max)	<b>94</b>	30%
Velocidad viento	<b>45</b>	14%
Humedad	<b>56</b>	18%
Probabilidad de precipitaciones	<b>87</b>	28%
Presión	<b>22</b>	7%
Otras (si consideró esta opción por favor sugerirla al final de la encuesta)	<b>10</b>	3%

Figura 2.10: Pregunta 10.

Analizando la figura 2.10 se puede notar lo siguiente:

- Se puede concluir que las características que más interesan son la temperatura seguido por la probabilidad de precipitaciones, luego en menor medida la humedad, velocidad del viento.

## 2.4. Sugerencias y comentarios

Dentro de los comentarios dejados por los participantes de la encuesta se encuentran las siguientes:

- Que la aplicación sea intuitiva.
- Que muestre las temperaturas por cada hora del día, o cada cierto intervalo de horas.

- Determinar si estará soleado o nublado.
- Pronóstico para más de un día.

## Anexo 3

# Especificación de pruebas

### 3.1. Prueba bloque 1

ID caso de prueba	Características a probar	Datos de entrada	Salida esperada	Salida obtenida	Éxito / Fracaso
		Nombre comuna			
conex1	Solicitud de pronóstico (con conexión a internet).	coronel	(array de tipo Json con la información)	(array de tipo Json con la información)	Éxito
conex2	Solicitud de pronóstico (sin conexión a internet).	coronel	mensaje de error de conexión.	mensaje de error de conexión.	Éxito

Tabla 3.1: Prueba bloque ID 1.

**Fuente:** Elaboración propia.

### 3.2. Prueba bloque 2

ID caso de prueba	Características a probar	Datos de entrada		Salida esperada	Salida obtenida	Éxito / Fracaso
map1	Correcta visualización de elementos en el mapa.	Interacción del usuario con la interfaz gráfica (mapa).		Nueva figura en el mapa señalando la comuna pulsada.	Nueva figura en el mapa señalando la comuna pulsada.	Éxito.
map2	Correcto despliegue de información de pronóstico.	Respuesta del servidor (Array de tipo Json).		visualización de la información de manera correcta.	visualización de la información de manera correcta.	Éxito.

Tabla 3.2: Prueba bloque ID 2.

**Fuente:** Elaboración propia.

### 3.3. Prueba bloque 3

ID caso de prueba	Características a probar	Datos de entrada		Salida esperada	Salida obtenida	Éxito / Fracaso
calc1	Correcto cálculo de dirección del viento.	Variables de viento vectoriales: $i = 4, j = 4$ .	NE (NorEste)	NE	Éxito	
calc2	Correcto cálculo de temperaturas mínima, máxima.	Lista de temperaturas del archivo seleccionado, para la prueba con la comuna de coronel.	Correctos valores calculados.	Correctos valores calculados.	Éxito	
calc3	Correcto cálculo de la humedad.	Valores asociados a la formula para obtener la humedad relativa.	Correcto valor calculado.	Correcto valor calculado.	Éxito	

Tabla 3.3: Prueba bloque ID 3.

**Fuente:** Elaboración propia.

### 3.4. Prueba bloque 4

ID caso de prueba	Características a probar	Datos de entrada	Salida esperada	Salida obtenida	Éxito / Fracaso
mat1	Correcta creación de archivos.	Para cada comuna se ingresa el nombre que tendrá el archivo asociado, además de la latitud y longitud del punto donde se realiza la muestra del modelo.	Archivo de texto con toda la información necesaria para realizar el pronóstico.	Archivo de texto con toda la información necesaria para realizar el pronóstico.	Éxito

Tabla 3.4: Prueba bloque ID 4.

**Fuente:** Elaboración propia.