Arquitectura de Software

Proyecto **AgroCheck**



**Integrantes:**

Nickolas Torres, Nickolas.torres@alumnos.ucm.cl

José Castillo, jose.castillo@alumnos.ucm.cl

Nicolas Arellano, Nicolas Arellano01@alumnos.ucm.cl

Contenido

[**1. Introducción 3**](#_sy6dpk56t9bu)

[1.1 Propósito 3](#_upqkc4cz4t7f)

[1.2 Alcance 3](#_edakmsd3r95p)

[1.3 Definiciones, acrónimos y abreviaturas 4](#_69b4ef6lfdpn)

[1.4 Referencias 4](#_4i5xypco1udm)

[**2. Representación arquitectural 5**](#_ksh13doxwq8z)

[2.1 Metas y limitaciones arquitectónicas 5](#_nwzm3dff9rxa)

[2.2 Vista de casos de uso 5](#_qcu0aftzyqex)

[2.3 Descripción general de la arquitectura: capas de paquetes y subsistemas. 7](#_5ef7i494tr1p)

[2.4 Vista del proceso 9](#_bx0vq5bq4m5p)

[2.5. Vista despliegue 10](#_xtkzgae3z7si)

# **1. Introducción**

Este documento presenta la información de la Arquitectura de Software Agrícola llamado Agrocheck. El proyecto tiene como propósito desarrollar una aplicación integral para la gestión eficiente de los recursos humanos y la supervisión de las actividades agrícolas. Esta aplicación permitirá registrar y mantener información actualizada sobre los trabajadores, así como gestionar las entradas de los cosechadores. A través de esta herramienta, se busca optimizar la administración del personal agrícola, mejorar el seguimiento de las actividades diarias y asegurar una gestión más efectiva de los recursos. El sistema estará diseñado para ser intuitivo y fácil de usar, facilitando la integración y el flujo de información entre los distintos módulos que componen la solución

## 1.1 **Propósito**

El propósito del proyecto es desarrollar una aplicación integral para la gestión eficiente de los recursos humanos y la supervisión de las actividades agrícolas. Esta aplicación permitirá registrar y mantener información actualizada sobre los trabajadores, así como gestionar las entradas de los cosechadores.

## 1.2 **Alcance**

Este documento de arquitectura de software proporciona una descripción general de la arquitectura del sistema. El sistema está siendo desarrollado por Agrocheck.Technology para apoyar al cosechador agrícola , También el alcance del proyecto se extiende a lo largo de todas las etapas del ciclo de vida del software, abarcando desde la planificación inicial hasta la implementación y mantenimiento continuo. Incluye el diseño detallado de la arquitectura de la aplicación, la definición de los requisitos funcionales y no funcionales, así como la selección de las tecnologías adecuadas para su desarrollo. Se contempla la creación de módulos específicos para la gestión de trabajadores y registros de entradas de cosechadores, asegurando la integración fluida entre ellos.

El alcance del proyecto abarca todas las actividades necesarias para desarrollar, implementar y mantener una solución integral para la gestión de recursos humanos y actividades agrícolas en un entorno eficiente y productivo.

## 1.3 **Definiciones, acrónimos y abreviaturas**

Las definiciones, acrónimos y abreviaturas son las siguientes

| 1 | CRUD | C=Ingresar, R=Consultar, U=Actualiza y D=borrar. |
| --- | --- | --- |
| 2 | DBMS | Sistema de Gestión de Bases de Datos (Database Management System). |
| 3 | ERP | Planificación de Recursos Empresariales (Enterprise Resource Planning). |
| 4 | KPI | Indicadores Clave de Rendimiento (Key Performance Indicators). |

## 1.4 **Referencias**

Las referencias son las siguientes:

* *ISO/IEC 9126-1:2001. Information technology - Software product quality - Part 1: Quality model. International Organization for Standardization.*
* *IEEE Std 830-1998. IEEE Recommended Practice for Software Requirements Specifications. Institute of Electrical and Electronics Engineers.*
* *Pressman, R. S. (2014). Software Engineering: A Practitioner's Approach. 8th Edition. McGraw-Hill Education.*
* *Descripción del Proyecto y Alcance, 2024, Agrocheck Project.*

Especificaciones del cliente: Se han considerado los requerimientos y especificaciones proporcionados por el cliente en relación con las funcionalidades y características esperadas del sistema.

·

# **2. Representación arquitectural**

Este documento presenta la arquitectura como una serie de vistas; vista de casos de uso, vista lógica, vista de procesos y vista de implementación. No hay una vista de implementación separada descrita en este documento. Estas son opiniones sobre un modelo subyacente de Lenguaje de modelado unificado (UML).

## **2.1 Metas y limitaciones arquitectónicas**

Existen algunos requisitos clave y limitaciones del sistema que tienen una influencia significativa en la arquitectura. Ellos son:

* La aplicación debe ser fácil de usar y tener una interfaz de usuario intuitiva
* la aplicación pueda manejar eficientemente un gran volumen de datos. Para lograrlo, es fundamental implementar una arquitectura escalable y robusta que pueda gestionar la carga de trabajo sin comprometer el rendimiento
* Después de ingresar al sistema (con login y password) el sistema realiza control de sesión sobre todas las operaciones realizadas por el usuario.
* la aplicación tiene que funcionar de manera adecuada en diversos dispositivos móviles y sistemas operativos es crucial para ofrecer una experiencia de usuario consistente y satisfactoria
* Cada requisito tiene un único identificador

## **2.2 Vista de casos de uso**

Operador

* Agregar trabajador: El operador puede agregar nuevos cosechadores(trabajador) con el nombre y el rut vinculándolo a un QR.
* Vincular código QR: El operador puede seleccionar un QR y vincularlo a un cosechador nuevo, de igual manera puede reutilizar un QR de otro cosechador que ya no esté.
* Desvincular código QR: El operador puede desvincular el QR de un cosechador para volver a utilizarlo y vincularlo con otro cosechador.
* Ingresar entrega cosechador: El cosechador puede ingresar el número de entregas realizadas por cosechador.
* Editar cosechador:: él puede editar el nombre y el rut del cosechador en un apartado diferente en el móvil según cada cosechador.

Administrador

* Crear operador: El administrador puede crear un operador con la información rut, nombre, contraseña.
* editar operador: El administrador puede editar operadores con la información rut, nombre, contraseña.
* eliminar operador: El administrador puede eliminar operadores existentes con respecto a su id.
* listar operador: El administrador puede visualizar los operadores existentes.
* Generar análisis gráfico:se generan gráficos sobre los datos de recolecta de los cosechadores, cantidades de bienes, el rendimiento de los trabajadores, etc.

Aca esta la estructura del Agrocheck considera distintos subsistemas relacionados entre sí, los cuales son nombrados a continuación:

Control de acceso : Controla el acceso al sistema a través de el rut y password para el administrador, con rut para el cosechador y rut y contraseña única para el operador. Después del ingreso se controla la ejecución de operaciones a través de control de sesión y control de permiso.

Ubicación : Permite al administrador manejar todas las acciones de un CRUD sobre la ubicación.

Frutas : Permite al administrador manejar todas las acciones de un CRUD sobre las frutas .

Cosechador : Permite al operador manejar las acciones de un CRU sobre el cosechador, al administrador R de cosechadores.

Operador : Permite al administrador manejar las acciones de un CRUD sobre el operador.

Bines : Permite al operador manejar las acciones de CR sobre los bines , permite al administrador la acción de R sobre los bines.

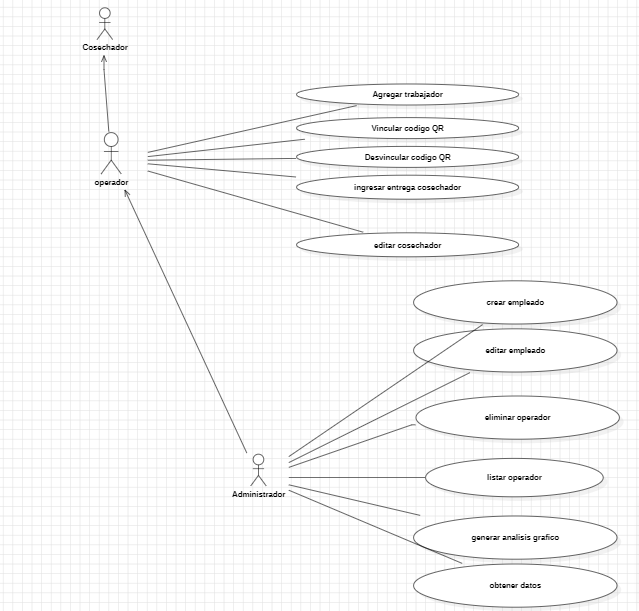
cantidad cosechada : Permite al operador manejar las acciones de CR sobre la cantidad cosechada , permite al administrador la acción de R sobre la cantidad cosechada con los datos del cosechador y al cosechador pues ver sus datos.

QR : Permite al administrador manejar las acciones de CR sobre los QR.

Visualización de análisis eficiencia y tendencia : Permite al administrador manejar las acciones de R sobre el análisis eficiencia y tendencia.

Visualización de gráficos. Incluye diferentes datos sobre las estadísticas de rendimiento y cambio de cosecha de los cosechadores para el administrador y para el cosechador ver el como va.

Informe análisis eficiencia y tendencia : Permite al administrador manejar las acciones de descargar información sobre el análisis eficiencia y tendencia.



## **2.3 Descripción general de la arquitectura: capas de paquetes y subsistemas.**

La arquitectura del proyecto AgroCheck está diseñada en varias capas, cada una con su propio conjunto de responsabilidades y funcionalidades. A continuación, se describe la estructura general de las capas de paquetes y subsistemas del proyecto.

1. **Capa de Presentación (Frontend)**

Vista Web:

* HTML: Estructura básica del contenido web.
* CSS: Estilo y diseño de la interfaz de usuario.
* JavaScript: Comportamiento dinámico y lógica de cliente.

Vistas Móvil:

* Interfaces de Usuario: Diseños y componentes específicos para dispositivos móviles.
* Controladores Móviles: Lógica de control específica para la aplicación móvil.

1. **Capa de Aplicación (Lógica de Negocio)**

Esta capa contiene la lógica de negocio de la aplicación. Se encarga de procesar las peticiones del usuario, aplicar las reglas de negocio y gestionar las interacciones entre la capa de presentación y la capa de datos.

Paquetes y Subsistemas:

Controladores:

* Controladores Web: Gestionan las peticiones HTTP y dirigen las respuestas a las vistas adecuadas.
* Controladores de API: Manejan las peticiones API desde las vistas y otros servicios.

Servicios:

* Servicios de Negocio: Implementan la lógica de negocio principal.
* Servicios de Integración: Gestionan la comunicación con sistemas externos y servicios de terceros.

1. **Capa de Datos**

Esta capa se encarga de la gestión de los datos persistentes de la aplicación. Incluye el acceso a la base de datos y otras formas de almacenamiento persistente.

Repositorios:

* Repositorios de Entidades: Gestionan la persistencia de las entidades de negocio.
* Repositorios de Consultas: Proveen métodos para recuperar datos de manera eficiente.
* Mapeo de Entidades: Definiciones de las entidades y su mapeo a las tablas de la base de datos.

1. **Capa de Integración**

Esta capa maneja la comunicación y la integración con otros sistemas y servicios externos.

API Gateway:

* Puerta de Enlace: Centraliza las peticiones API y las distribuye a los microservicios correspondientes.

Adaptadores de Integración:

* Conectores de Servicios Externos: Gestionan la comunicación con servicios de terceros y otros sistemas.

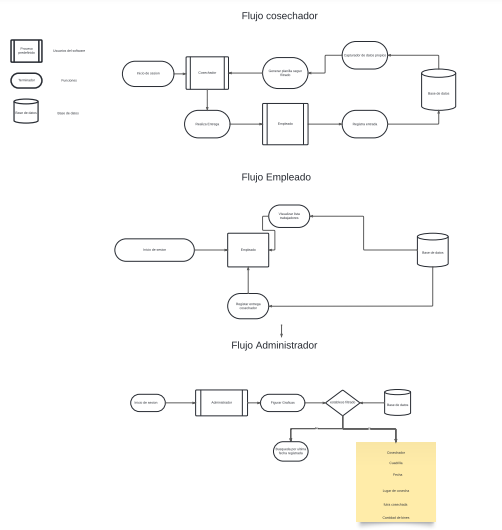
## **2.4 Vista del proceso**

El proceso de ejecución normal del sistema con los 3 flujos o procesos relacionados con diferentes roles dentro de una organización o sistema.

Flujo cosechador: Representa el proceso para los trabajadores encargados de la cosecha

Flujo Empleado: Describe el flujo de trabajo para los empleado

Flujo Administrador: Este flujo está destinado a los administradores



## 

## 

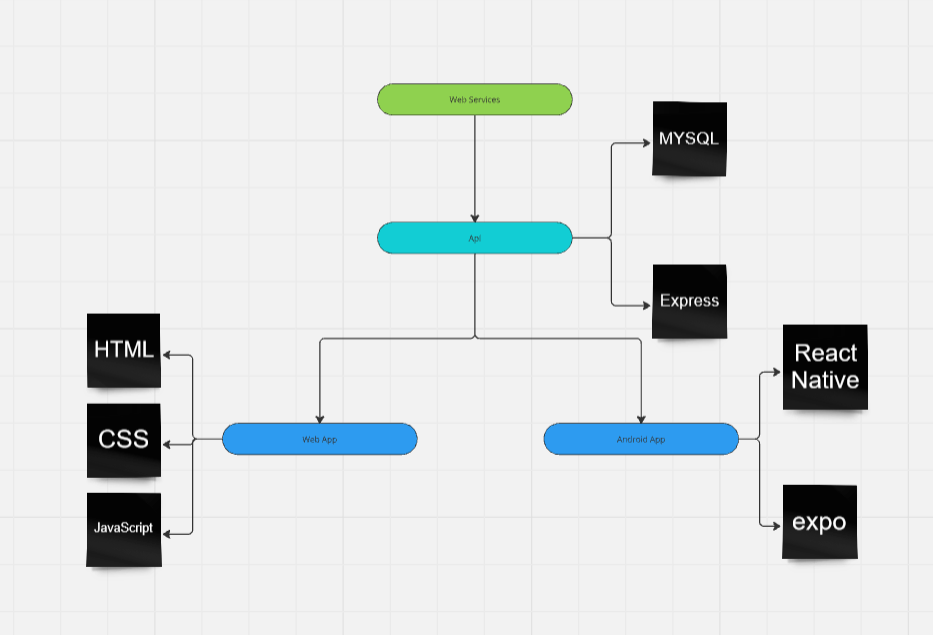
## 

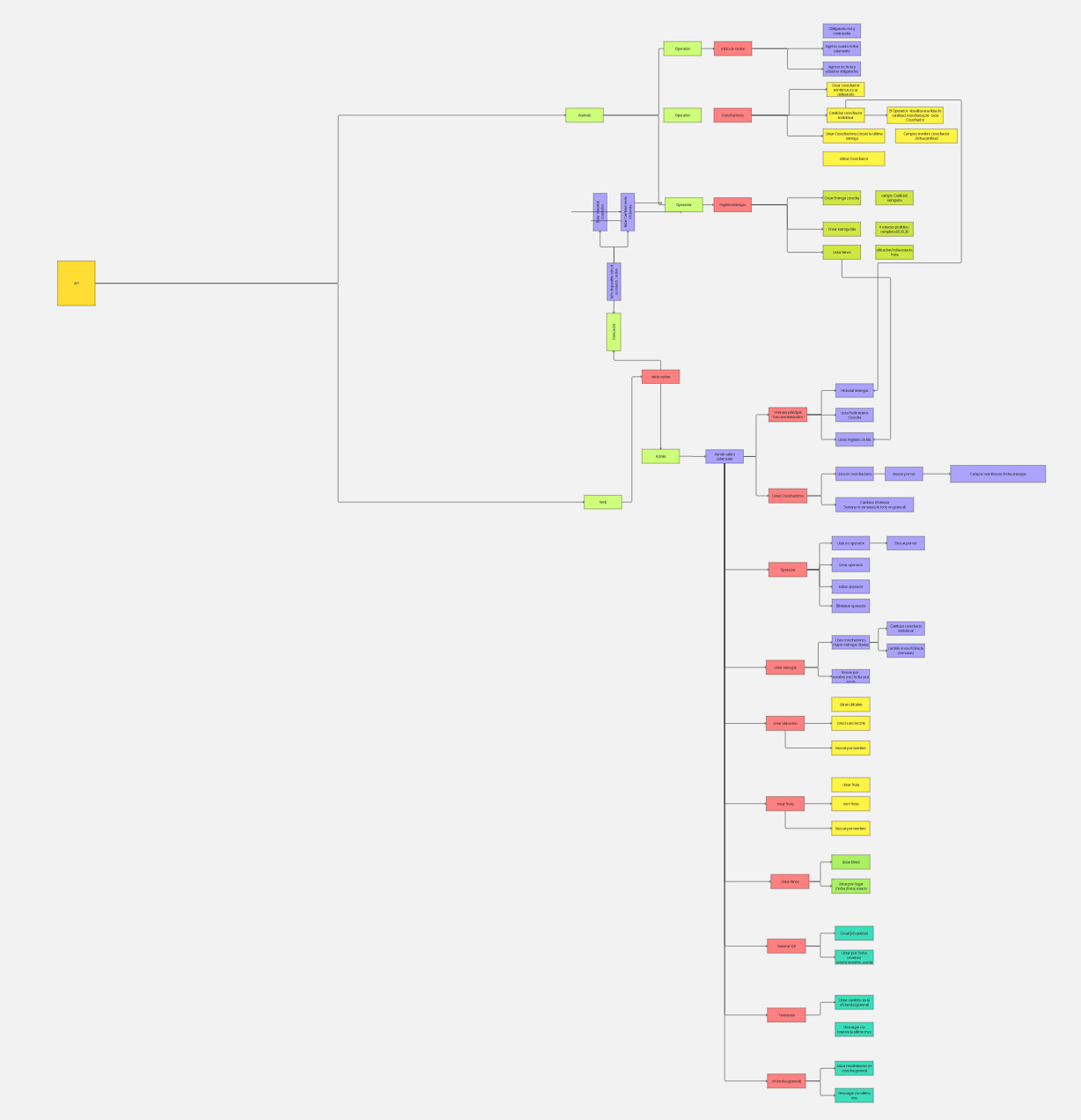
## 

## **2.5.**

## **Vista despliegue**

En la ejecución del sistema tenemos la ejecución de vista html, css, javascript; tras esto debemos de recordar que tenemos el apartado móvil y el apartado web. en donde la vista web se conecta con la api del back-end para hacer peticiones, logrando crear, listar, editar y eliminar el contenido.

****

**Vista Api**  
Link de vista completa   
<https://miro.com/welcomeonboard/Z0dwaWRtdXpaQVdjTW5RMUZhQ1lMSU9SNGJMcFJOSHVnTUpReHhMMzJ6ZFg1UUtFMlNBcmlHVUJKYlByWERMa3wzNDU4NzY0NTg1NTA1MjcwODgyfDI=?share_link_id=230419079069>

vista Base de datos  
  
