| DUOC UC - Escuela de informática y telecomunicaciones |
| --- |
| Propuesta de Proyecto y Especificación de Requisitos de Software |
| *Proyecto: [Duoc OfCourse]* |
|  |
| **Revisión*: [01]*** |
| **[02/10/2025]** |

| Planificación y Especificación de Requisitos según estándares; IEEE 830, ISO9000 y PMI. |
| --- |

**Contenido**

[*DUOC UC - Escuela de informática y telecomunicaciones 1*](#_heading=h.xvahlha85lts)

[**Ficha del documento 4**](#_heading=h.e0jz2jjtye57)

[**1. Introducción 5**](#_heading=h.wf1rvwjubz0o)

[1.1.](#_heading=h.q9k0qw6ww0uf) Propósito 5

[1.2.](#_heading=h.mv5s8usdlnew) Ámbito del Sistema 5

[1.3.](#_heading=h.2xzrxup7qzqe) Definiciones, Acrónimos y Abreviaturas 5

[1.4.](#_heading=h.w59iw01o5nwq) Referencias 5

[1.5.](#_heading=h.osmwbhux0e30) Visión General del Documento 5

[**2.**](#_heading=h.x4o2v5ixonhd) **Descripción General 6**

[2.1.](#_heading=h.a6gqjqcqcmyt) Perspectiva del Producto 6

[2.2.](#_heading=h.yv8rkcrzu9jo) Funciones del Producto 6

[2.3.](#_heading=h.yqz5b329mywl) Características de los Usuarios 6

[2.4.](#_heading=h.w008oax37dw9) Restricciones 6

[2.5.](#_heading=h.i7a4fqfblpv6) Suposiciones y Dependencias 7

[2.6.](#_heading=h.vvzkvyefzkgp) Requisitos Futuros 7

[**3.**](#_heading=h.nuwppsyq4rjn) **Requisitos Específicos 8**

[3.1](#_heading=h.zz5rf3gmvs3) Requisitos comunes de las interfaces 8

[*3.1.1*](#_heading=h.86dc5hjos7cl) *Interfaces de usuario 8*

[*3.1.2*](#_heading=h.rehtbll7x9xr) *Interfaces de hardware 8*

[*3.1.3*](#_heading=h.dj5aoc81gkb7) *Interfaces de software 8*

[3.2](#_heading=h.w3snrrwzptre) Requisitos funcionales 9

[3.3](#_heading=h.ufuag0elejwa) Requisitos no funcionales (Organización) y de calidad (Producto) 9

[*3.3.1*](#_heading=h.r18be3tj2sgp) *Requisitos de Rendimiento o Eficiencia 9*

[*3.3.2*](#_heading=h.pyuex9ybrqc2) *Requisitos de Seguridad 9*

[*3.3.3*](#_heading=h.w587bnst6umk) *Requisitos de Usabilidad 9*

[*3.3.4*](#_heading=h.6mnrx8lh36yt) *Requisitos de Disponibilidad 9*

[*3.3.5*](#_heading=h.59xcx91n3xz2) *Requisitos de Portabilidad 9*

[*3.3.6*](#_heading=h.dt2bzjmpqpdu) *Requisitos de Mantenibilidad 9*

[*3.3.7*](#_heading=h.v04aoo8w4nsz) *Requisitos de Funcionalidad 9*

[3.4](#_heading=h.xu4iudosf2ya) Requisitos No funcionales Organizacionales 10

[**4. Propuesta de Planificación 11**](#_heading=h.knt6zhkavdhj)

[4.1 Descripción general acerca de la Planificación 11](#_heading=h.eetlb1v2qs55)

[*4.1.2 Definición del Equipo de Trabajo 11*](#_heading=h.nvu2jlhn48qo)

[*4.1.3 Definición de Actividades principales del Proyecto 11*](#_heading=h.gov1u82z4u9a)

[*4.1.4 Resumen Costos del Desarrollo del Proyecto 11*](#_heading=h.h425kin1u6u0)

[*4.1.6 Carta Gantt 11*](#_heading=h.xwtacvm5g2xw)

[5. Anexos 12](#_heading=h.qzyum070dc0h)

[*5.1 Acta de Proyecto 12*](#_heading=h.j9rftnowqk1h)

[*5.2 Matriz Especificación de Requerimientos 12*](#_heading=h.us7orjjiwp5k)

[*5.3 Prototipado de Software 12*](#_heading=h.rexyljrgl0df)

[*5.4 Matriz EDT. Planilla Detallada Cálculo de Esfuerzo 12*](#_heading=h.4z7qtq4i3j5q)

[*5.5 Planilla Carta Gantt 12*](#_heading=h.2y31xdhzte7o)

# Ficha del documento

| **Fecha** | **Revisión** | **Autor** | **Modificación** |
| --- | --- | --- | --- |
| *07-09-2025* | *1.0* | *Nicolás Canales* | *Inicio del documento.* |
| *10-09-2025* | *1.1* | *Benjamín Gamboa* | *Ámbito del sistema.* |
| *10-09-2025* | *1.2* | *Nicolás Canales* | *Propósito.*  *Funciones del sistema.*  *Características de usuarios.*  *Restricciones.* |
| *19-09-2025* | *1.3* | *Nicolás Canales* | *Resumen costos del desarrollo.*  *Definición del equipo de trabajo.* |
| *29-09-2025* | *1.4* | *Nicolás Canales* | *Inicio interfaces de software.*  *Adición de requerimiento funcional.* |
| *30-09-2025* | *1.5* | *Benjamín Guajardo* | *Definiciones, acrónimos y abreviaturas.*  *Visión general del documento.*  *Suposiciones y Dependencias.*  *Interfaces de hardware.*  *Requisito de usabilidad*  *Requisitos de portabilidad* |
| *01-10-2025* | *1.6* | *Nicolás Canales* | *Fin de interfaces de software.* |

Documento validado por las partes en fecha:

**Integrantes:**

| **Nombre Integrante del Equipo** | **Rol Definido** |
| --- | --- |
| *Nicolás Canales* | *Jefe de proyecto.* |
| *Benjamín Gamboa* | *Desarrollador Backend* |
| *Benjamín Guajardo* | *Desarrollador Frontend - DBA* |
|  |  |
|  |  |

# 1. Introducción

## 1.1. Propósito

La finalidad de este documento es definir y ordenar de manera clara los requerimientos del sistema, tanto funcionales como no funcionales, costos y fechas clave, para asegurar que todos los integrantes del equipo de trabajo tengan claro el funcionamiento y los objetivos del sistema. Este documento va dirigido al equipo de trabajo e interesados.

¿Qué hará?

* Gestión de cursos, exámenes y certificaciones.
* Realización y seguimiento de progreso de cursos.
* Notificaciones por correo electrónico.
* Gestión de usuarios.
* Gestión de sesión.
* Generación de documentos.

¿Qué no hará?

* Reemplazar plataformas existentes de Duoc UC.

Beneficios y objetivos:

* Implementar dos plataformas web que se comuniquen entre sí para la realización de cursos en línea que ofrezca Duoc.
* Mejorar la experiencia de aprendizaje y seguimiento académico de los estudiantes.
* Reducir tiempos de administración y generación de certificados.
* Garantizar la seguridad de los usuarios y el cumplimiento de normativas de privacidad.

## 1.2. Ámbito del Sistema

DuocOfCourse buscará ser un apoyo a la comunidad estudiantil de las sedes de duoc, no solo consta con cursos de nuestra escuela sino que todo alumno sin importar su escuela será bienvenido a aportar en esta App.

El objetivo principal de DuocOfCourse es ofrecer cursos de todas las carreras para los alumnos logueados en el sistema, un alumno de informática podrá aprender fundamentos de mecánica automotriz y así con todas las carreras que existen hoy en Duoc.

Una vez finalizada la construcción de esta app, ésta estará a libre disposición para los estudiantes, en conjunto con los PDF, material de lectura y actividades entregadas por los profesores.

## 

## 

## 1.3. Definiciones, Acrónimos y Abreviaturas

* **API**: Interfaz de Programación de Aplicaciones, un conjunto de definiciones y protocolos que permiten que una aplicación interactúe con otra.
* **RESTful**: Estilo arquitectónico para servicios web que utiliza los métodos HTTP para permitir la comunicación entre sistemas.
* **XLS**: Formato de archivo de hojas de cálculo de Microsoft Excel, utilizado para almacenar datos tabulares.
* **PDF**: Formato de documento portátil utilizado para representar documentos independientemente del software, hardware o sistema operativo.
* **Frontend**: Parte del desarrollo que interactúa directamente con el usuario. En este caso, se refiere a la interfaz de usuario de la plataforma (desarrollada con React).
* **Backend**: Parte del desarrollo que maneja la lógica y las operaciones en el servidor. En este caso, se encarga de gestionar la base de datos y las operaciones del sistema.
* **Base de Datos (MySQL)**: Sistema de gestión de bases de datos utilizado para almacenar la información del sistema, como datos de usuarios, cursos y exámenes.
* **Login/Autenticación**: El proceso mediante el cual los usuarios ingresan al sistema con su cuenta personal (correo y contraseña) para acceder a las funciones del sistema.
* **UI**: (User Interface) Interfaz de usuario, el diseño visual con el que los usuarios interactúan dentro de la aplicación.
* **UX**: (User Experience) Experiencia del usuario, el proceso general de interactuar con la plataforma desde el punto de vista del usuario.
* **CRUD**: Operaciones básicas realizadas en bases de datos: Crear, Leer, Actualizar, Borrar.
* **MVP**: Producto Mínimo Viable, una versión de un producto con las características mínimas necesarias para ser funcional, lanzado con el objetivo de obtener feedback.
* **JWT**: JSON Web Token, un estándar abierto utilizado para compartir datos de manera segura entre dos partes, comúnmente usado en sistemas de autenticación.
* **OAuth2**: Protocolo estándar para autorización que permite acceso seguro a las aplicaciones, utilizado para la autenticación y autorización de usuarios.  
  **Duoc UC**: Universidad Técnica de Chile, institución educativa en la que se desarrolla este proyecto.
* **AVA**: Ambiente Virtual de Aprendizaje, plataforma digital utilizada por Duoc UC para gestionar el aprendizaje en línea.
* **HTTP/HTTPS**: Protocolo de transferencia de hipertexto (Hypertext Transfer Protocol/Secure), utilizado para la comunicación en la web.
* **HTML**: Lenguaje de marcado utilizado para estructurar contenido en la web.
* **CSS**: Lenguaje de hojas de estilo utilizado para describir la presentación de un documento HTML.
* **JavaScript**: Lenguaje de programación utilizado para crear interactividad en la web, especialmente en el frontend.
* **MySQL**: Sistema de gestión de bases de datos relacional, utilizado para almacenar datos en este sistema.
* **React**: Biblioteca de JavaScript para construir interfaces de usuario, utilizada para el desarrollo del frontend.
* **.NET**: Framework de desarrollo de software para construir aplicaciones web, utilizado para el desarrollo del backend del sistema.

## 1.4. Referencias

[DOCUMENTOS DEL DRIVE.](https://drive.google.com/drive/folders/11yhxw4CC3Kndmn0SjFp9qQ6MriIokEz8?usp=drive_link)

## 1.5. Visión General del Documento

Este documento está estructurado de manera que cubre todos los aspectos clave del proyecto, desde la definición del sistema hasta los requisitos específicos y la planificación de la implementación. A continuación, se describe brevemente el contenido de cada sección:

**Sección 1: Introducción**

En esta sección se proporciona el propósito y el alcance del proyecto, así como las definiciones clave, acrónimos y abreviaturas utilizados en todo el documento. También se detallan las referencias y la visión general de lo que se cubrirá en el documento.

**Sección 2: Descripción Genera**l

Aquí se describe el sistema en términos generales, incluyendo su relación con otros productos o sistemas existentes, las funciones principales, las características de los usuarios y las restricciones del sistema. También se abordan las suposiciones y dependencias que podrían afectar al desarrollo del sistema.

**Sección 3: Requisitos Específicos**

Esta es la sección central del documento, donde se detallan los requisitos funcionales y no funcionales del sistema. Se incluyen los requisitos para las interfaces de usuario, hardware y software, así como los requisitos de rendimiento, seguridad, usabilidad, disponibilidad, portabilidad, mantenibilidad y funcionalidad.

**Sección 4: Propuesta de Planificación**

En esta sección se proporciona un resumen de la planificación del proyecto, incluyendo la descripción general del proceso de trabajo, la definición del equipo de trabajo, las actividades principales y los costos asociados al desarrollo. También se presenta una carta Gantt que establece las fechas clave del proyecto.

**Sección 5: Anexos**

Los anexos contienen documentos adicionales que respaldan el proyecto, como el Acta de Proyecto, la Matriz de Especificación de Requerimientos, los prototipos de software, la Matriz EDT y la carta Gantt detallada.

# 2. Descripción General

## 2.1. Perspectiva del Producto

Esta subsección debe relacionar el futuro sistema (producto software) con otros productos. Si el producto es totalmente independiente de otros productos, también debe especificarse aquí. Si la ERS define un producto que es parte de un sistema mayor, esta subsección relacionará los requisitos del sistema mayor con la funcionalidad del producto descrito en la ERS, y se identificarán las interfaces entre el producto mayor y el producto aquí descrito.

Se propondrá la integración del Login para ambas páginas con el API de Duoc UC, para ocupar las cuentas creadas con los correos institucionales de Duoc UC.

## 2.2. Funciones del Producto

-El sistema Duoc OfCourse tendrá dos aplicaciones principales: Una para docentes y administradores y otra para estudiantes. Las funciones de cada plataforma estarán orientadas a mejorar la gestión de cursos online, el seguimiento del aprendizaje y la interacción entre los usuarios.

Funciones principales:

* DuocOfCourseDocentes:

1. Gestión de cursos: Crear, modificar y eliminar cursos.
2. Gestión de exámenes: Crear y administrar exámenes y evaluaciones.
3. Generación de certificados: Se podrán generar certificados PDF y notas en XLS.

* DuocOfCourseEstudiantes:

1. Inscribir curso: Inscribirse a un curso del listado de cursos.
2. Realizar exámenes: Acceso a exámenes para concluir cursos.
3. Seguimiento de progreso: Visualizar el progreso del curso.
4. Notificar: Enviar notificaciones mediante correo electrónico.

* Ambas aplicaciones:

1. Interoperabilidad: Ambas aplicaciones están conectadas mediante una base de datos compartida.

## 2.3. Características de los Usuarios

**General:**

1. **Gestión de sesión:** Iniciar y cerrar sesión.

### Docente

* 1. **Características**:
     1. **Nivel Educacional**: Profesores de Duoc UC.
     2. **Experiencia Técnica**: Podría variar, pero se espera que tengan conocimientos básicos de herramientas digitales y plataformas educativas, como AVA o VivoDuoc.

1. **Estudiante**
   1. **Características**:
      1. **Nivel Educacional**: Estudiantes de nivel universitario en Duoc UC.
      2. **Experiencia Técnica**: Podría variar, pero se espera que tengan conocimientos básicos de herramientas digitales y plataformas educativas, como AVA o VivoDuoc.
2. **Administrador**:
   1. **Características**:
      1. **Nivel Educacional**: Profesionales con formación en áreas como administración, ingeniería o informática.
      2. **Experiencia Técnica:** Alto conocimiento técnico, especialmente en la gestión de sistemas y plataformas tecnológicas.

## 2.4. Restricciones

* Políticas de la empresa.
  + -El desarrollo del sistema debe alinearse con las políticas internas de **Duoc UC**, como la protección de datos, la accesibilidad y el cumplimiento de las normativas institucionales.
* Limitaciones del hardware.
  + El sistema debe estar optimizado para funcionar en los servidores disponibles en un servidor local.
* Interfaces con otras aplicaciones.
  + Debe tener un estilo similar al de las páginas oficiales de Duoc UC, además de asimilarse entre ambas.
* Operaciones paralelas.
  + El sistema debe ser capaz de manejar múltiples usuarios y solicitudes de forma simultánea, sin afectar el rendimiento. Especialmente la realización de los cursos y exámenes.
* Funciones de auditoría.
  + El sistema debe contar con registros de actividad (logs) para poder auditar el uso de la plataforma, especialmente en el caso de las interacciones con los exámenes y los certificados.
* Funciones de control.
  + Deben existir controles para verificar la autenticidad de los estudiantes y garantizar que solo los usuarios autorizados puedan acceder a ciertas funciones
* Lenguaje(s) de programación.
  + El sistema utilizará React para el frontend, .NET para el backend y MySQL para la base de datos.
* Protocolos de comunicación.
  + El sistema utilizará protocolos estándar de comunicación web como HTTP/HTTPS y APIs RESTful para la integración entre las aplicaciones docentes y estudiantiles.
* Requisitos de habilidad.
  + Los usuarios del sistema (docentes y estudiantes) deben tener un nivel básico de habilidades digitales para poder interactuar con la plataforma de manera efectiva. Mientras que los administradores deben tener un nivel avanzado de habilidades digitales y experiencia en gestión.
* Criticidad de la aplicación.
  + La plataforma será crítica para el desarrollo educativo de los estudiantes y docentes. Cualquier fallo en el sistema podría interrumpir el acceso a cursos o el seguimiento del progreso académico.
* Consideraciones acerca de la seguridad.
  + El sistema debe cumplir con las normativas de protección de datos como la encriptación de datos, y protección contra ataques cibernéticos.

## 2.5. Suposiciones y Dependencias

**Suposiciones:**

Usuarios con acceso a Internet: Se supone que todos los usuarios, tanto docentes como estudiantes, tendrán acceso a una conexión estable a Internet para acceder a la plataforma DuocOfCourse.

Disponibilidad de infraestructura: Se asume que la infraestructura tecnológica de Duoc UC (servidores, bases de datos, redes) será adecuada para soportar la implementación del sistema sin requerir modificaciones o mejoras adicionales.

Conocimiento básico en herramientas digitales: Se supone que los usuarios (docentes y estudiantes) tienen un conocimiento básico de herramientas digitales y plataformas educativas, como AVA o VivoDuoc.

Soporte para dispositivos y navegadores estándar: El sistema se desarrollará para ser compatible con los navegadores web más comunes (Google Chrome, Mozilla Firefox, Microsoft Edge) y dispositivos modernos, sin necesidad de realizar ajustes especiales en plataformas obsoletas o dispositivos muy antiguos.

**Dependencias:**

Integración con sistemas existentes: El sistema DuocOfCourse dependerá de la integración con el sistema de autenticación de Duoc UC, que utiliza correos institucionales para el acceso. Cualquier cambio en los servicios de autenticación de Duoc UC podría afectar al sistema.

Dependencia de APIs externas: El sistema puede depender de APIs externas para funciones como el envío de correos electrónicos de notificación y la generación de certificados en PDF. La disponibilidad y el rendimiento de estas APIs influirán directamente en el funcionamiento del sistema.

Cumplimiento de normativas de privacidad: El sistema está sujeto a las regulaciones locales sobre protección de datos personales (como la Ley 19.496 de Protección de Datos Personales en Chile). Los cambios en estas normativas podrían requerir ajustes en el diseño o funcionamiento del sistema.

Actualización de contenido: La disponibilidad y actualización del contenido académico (cursos, exámenes, materiales de lectura) dependerá de la colaboración de los docentes de Duoc UC. Cualquier retraso o falta de actualización de estos materiales afectará el progreso de los cursos y la disponibilidad de los mismos.

## 2.6. Requisitos Futuros

**Esta subsección esbozará futuras mejoras al sistema, que podrán analizarse e implementarse en un futuro.**

Se espera como mejora futura la creación de una interfaz de usuario propia, donde el usuario sea capaz de visualizar la información de su cuenta.

# 3. Requisitos Específicos

Esta sección contiene los requisitos a un nivel de detalle suficiente como para permitir a los diseñadores diseñar un sistema que satisfaga estos requisitos, y que permita al equipo de pruebas planificar y realizar las pruebas que demuestren si el sistema satisface, o no, los requisitos. Todo requisito aquí especificado describirá comportamientos externos del sistema, perceptibles por parte de los usuarios, operadores y otros sistemas. Esta es la sección más larga e importante de la ERS. Deberán aplicarse los siguientes principios:

• El documento debería ser perfectamente legible por personas de muy distintas formaciones e intereses.

• Deberán referenciarse aquellos documentos relevantes que poseen alguna influencia sobre los requisitos.

• Todo requisito deberá ser unívocamente identificable mediante algún código o sistema de numeración adecuado.

## 3.1 Requisitos comunes de las interfaces

Descripción detallada de todas las entradas y salidas del sistema de software.

### 3.1.1 Interfaces de usuario

* Las pantallas son de colores institucionales.
* Letra tamaño 10
* El color de las letras se definirá en base al color del fondo, para lograr un buen contraste y asegurar la correcta visualización de las letras.
* El progreso se mide en porcentaje.
* Las notas se miden en números con decimales (entre 1.0 como nota mínima y 7.0 como nota máxima).

### 3.1.2 Interfaces de hardware

Especificar las características lógicas para cada interfaz entre el producto y los componentes de hardware del sistema. Se incluirán características de configuración.

**Servidor de base de datos (MySQL)**:

* **Descripción**: El sistema se alojará en un servidor local, donde se instalará el servidor MySQL para la gestión de la base de datos.
* **Requerimientos**:
  + Capacidad de almacenamiento suficiente para manejar la información de los usuarios, cursos, exámenes y certificados.
  + Conexión de red estable para permitir el acceso simultáneo a la base de datos por múltiples usuarios.

**Dispositivos de usuario**:

* **Descripción**: Los usuarios accederán al sistema a través de dispositivos personales, como computadoras de escritorio, laptops o smartphones, utilizando navegadores web compatibles.
* **Requerimientos**:
  + **Mínimo**: Procesadores de 1.0 GHz, 2 GB de RAM y 20 GB de almacenamiento disponible.
  + **Recomendado**: Dispositivos con procesadores de 2.0 GHz o superior, 4 GB de RAM o más y pantallas con resolución de 1080p o superior.
  + Navegadores compatibles: Google Chrome, Mozilla Firefox, Microsoft Edge.

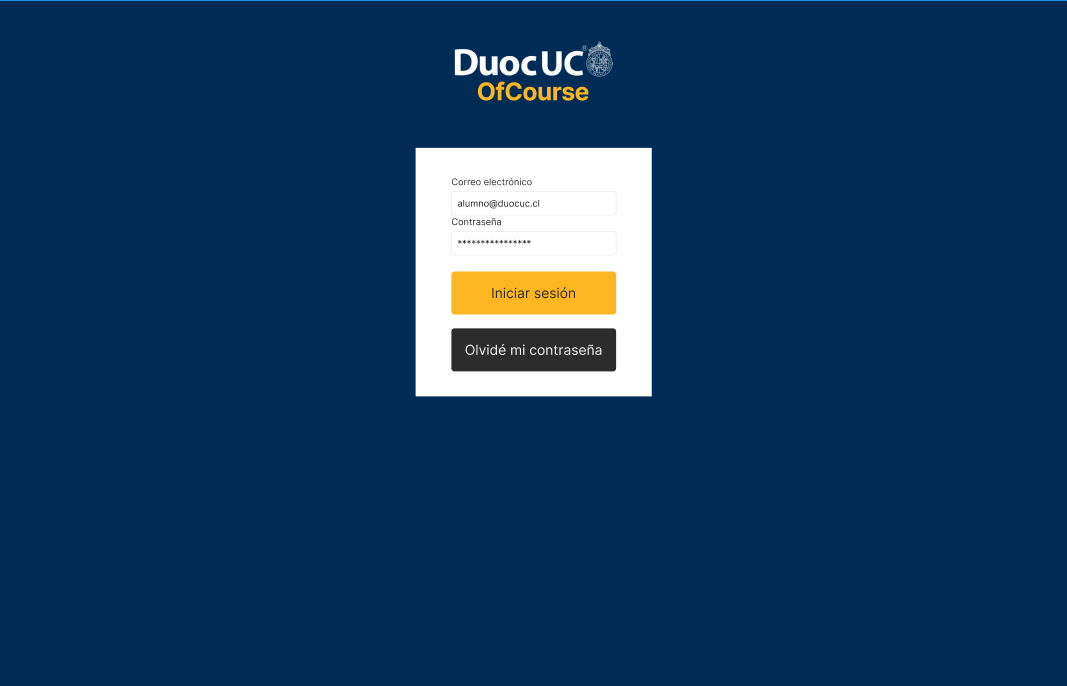
**Dispositivos de entrada/salida**:

* **Descripción**: Los usuarios interactúan con el sistema principalmente a través de teclados, ratones y pantallas táctiles en el caso de dispositivos móviles. Los dispositivos deberán ser capaces de soportar interfaces web responsivas.
* **Requerimientos**:
  + **Teclados y ratones**: Deben ser compatibles con las interfaces estándar de entrada de datos (USB o inalámbricos).
  + **Pantallas táctiles**: En dispositivos móviles y tabletas, las pantallas táctiles deben ser capaces de interactuar con formularios, botones y otros elementos de la interfaz de usuario.

### 3.1.3 Interfaces de software

**DuocOfCourseDocentes.**

**Interfaz de Login.**

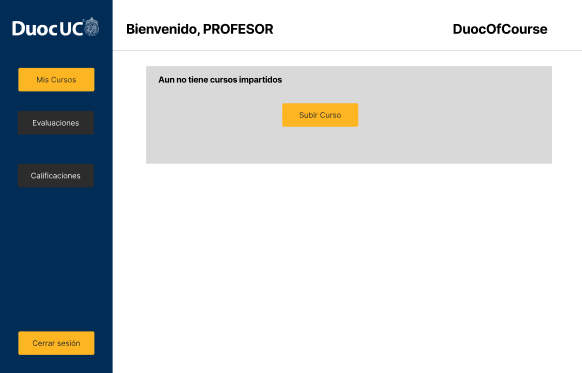
****

* **Descripción:** Primera interfaz que se visualiza al ingresar al sitio, se presenta un formulario para ingresar una cuenta, solicitando **correo electrónico** y **contraseña**.
* **Propósito:** Asegurar que no cualquier persona ingrese a la aplicación, además, al poseer sesiones distintas con roles, se asegura que cada usuario pueda realizar las tareas propias de su rol.
* **Definición del interfaz:** Se visualiza el nombre de la página: DuocOfCourse, debajo se presenta el formulario para ingresar el correo electrónico y la contraseña del usuario, debajo de este formulario se muestran dos botones: Iniciar sesión y Olvidé mi contraseña.

**Interfaz de Registro.**

* **Descripción:** Formulario a rellenar al seleccionar el botón de **Registrarse**.
* **Propósito:** Crear una cuenta para un usuario que no tenga una cuenta previamente.
* **Definición de interfaz:** Formulario que pide el correo electrónico, nombre completo, contraseña y carrera.

**Menú principal.**

****

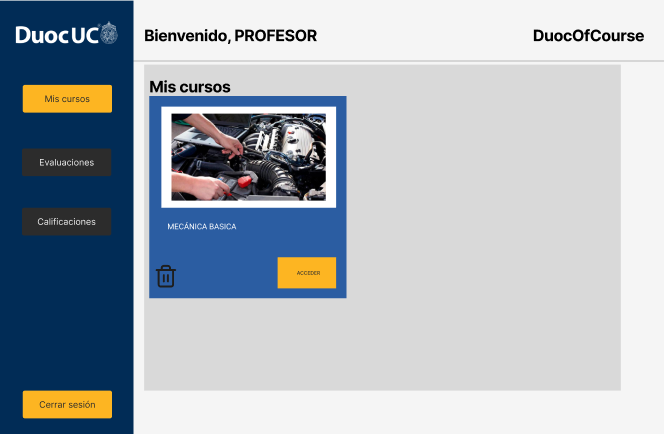
* **Descripción:** Menú al que se entra cuando se inicia sesión siendo **docente**, muestra los cursos que tiene y la navegación del docente.
* **Propósito:** Inicio de sesión correcto, opciones propias del **rol de docente**.
* **Definición del interfaz:** En este menú se visualizan los cursos que ha subido, además se muestra un botón para **Subir curso**, en el menú de la izquierda se ven las opciones de navegación del **docente**. Primero se ve **Mis cursos**, que muestra los cursos que tiene el docente, al final de este menú se muestra el botón para **Cerrar sesión**.

**Subir curso.**

****

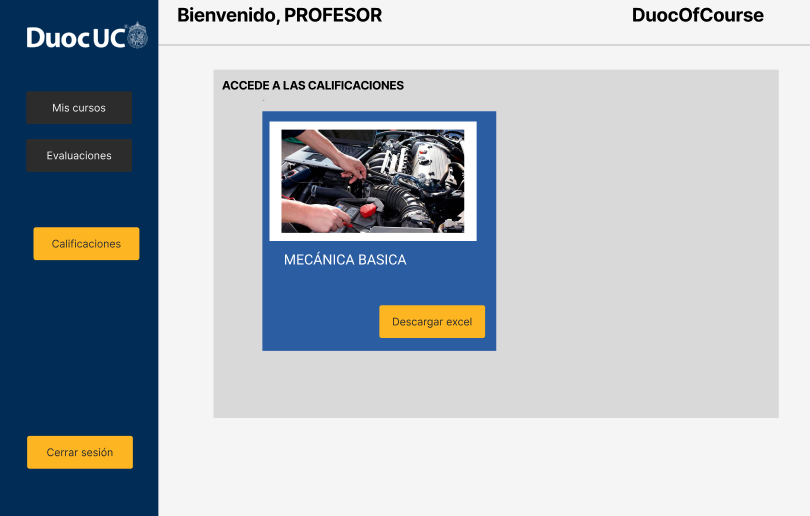
* **Descripción:** Formulario al que se ingresa mediante el botón de **Subir curso** del menú principal del docente.
* **Propósito:** La tarea principal del docente es la **gestión de cursos**, gracias a este formulario se pueden crear cursos para los estudiantes, que es la finalidad de la página.
* **Definición del interfaz:** Formulario para crear un curso, tiene tres secciones: Portada, Contenido y Resumen.
  + En la vista Portada se muestran los siguientes campos a rellenar:
    1. Título del curso.
    2. Descripción del curso.
    3. Portada del curso.
  + En la vista Contenido se muestran los siguientes campos a rellenar:
    1. Video del curso.
    2. Quiz del curso.
  + En la vista Resumen se muestra el resumen de la información previamente agregada en las vistas anteriores.

**Mis cursos.**

****

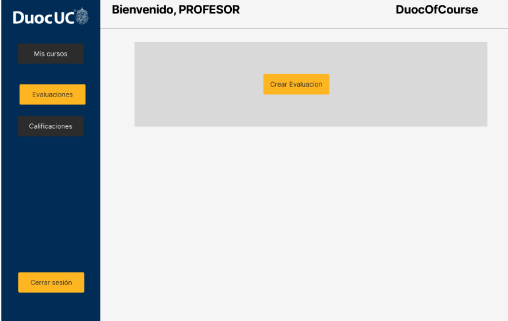
* **Descripción:** Vista para los docentes, se visualizan los cursos que ha creado el docente, aquí se presentan las opciones para modificar, borrar y crear cursos.
* **Propósito:** Que el docente gestione los cursos que creó o cree uno.
* **Definición del interfaz:** En esta interfaz se presenta un listado de cursos que creó el docente, cada uno de estos tiene un botón para acceder, un botón simbolizado por un Contenedor de basura para eliminar un curso y uno con un engranaje para modificarlos.

**Calificaciones.**

****

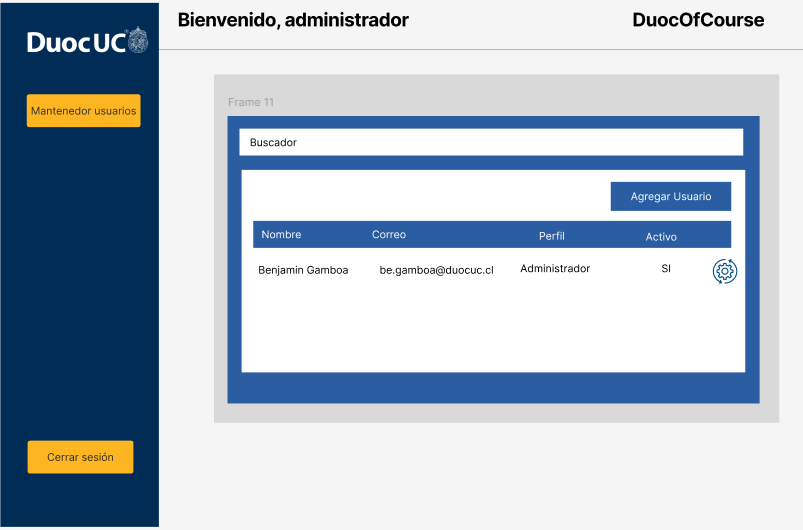
* **Descripción:** En esta interfaz el profesor puede ver los cursos que ha finalizado y los resúmenes de las calificaciones de cada curso..
* **Propósito:** Visualizar y descargar listado de calificaciones.
* **Definición del interfaz:** Se muestra un listado de cursos, cada curso tiene un botón **Descargar,** para que se descargue un documento Excel con el resumen de notas.

**Evaluaciones.**

****

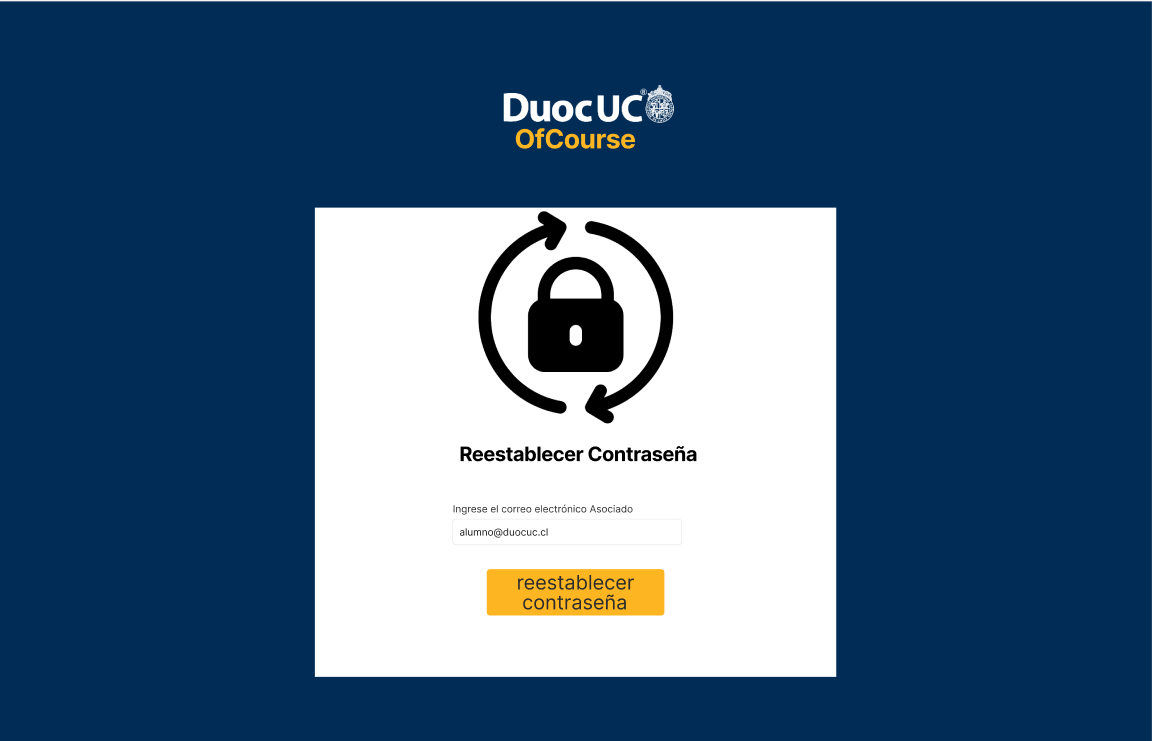
* **Descripción:** Vista donde se visualizan todos los exámenes finales para un curso, donde el docente puede crear un exámen final para cada curso.
* **Propósito:** Registrar y crear una evaluación para los estudiantes al final de un curso, este proceso es opcional para la certificación.
* **Definición del interfaz:** Se muestra el listado de exámenes y un botón para **Crear evaluación**, con esto se abre el formulario y se puede crear un exámen final.

**Modificar usuarios.**

****

* **Descripción:** Vista del administrador, aquí se listan los usuarios existentes, los usuarios pueden ser creados o modificados por el administrador.
* **Propósito:** Gestionar a los usuarios existentes.
* **Definición del interfaz:** Listado de usuarios, cada usuario tiene la opción de ser modificado, arriba del listado de usuarios se visualiza un botón que dice **Agregar usuario**, aquí se redirige al formulario para crear un usuario.

**Recuperar contraseña.**

****

* **Descripción:** Vista que se lleva desde la interfaz de Login, aquí se puede restablecer la contraseña si el usuario la ha olvidado o si desea cambiarla.
* **Propósito:** Cambiar la contraseña.
* **Definición del interfaz:** A través del botón de Olvidé mi contraseña, se muestra este breve formulario, que muestra el campo para escribir el correo electrónico de la cuenta y un botón de Restablecer contraseña, al completarlo se redirige a un mensaje que da aviso de un e-mail que se le mandó al correo ingresado.

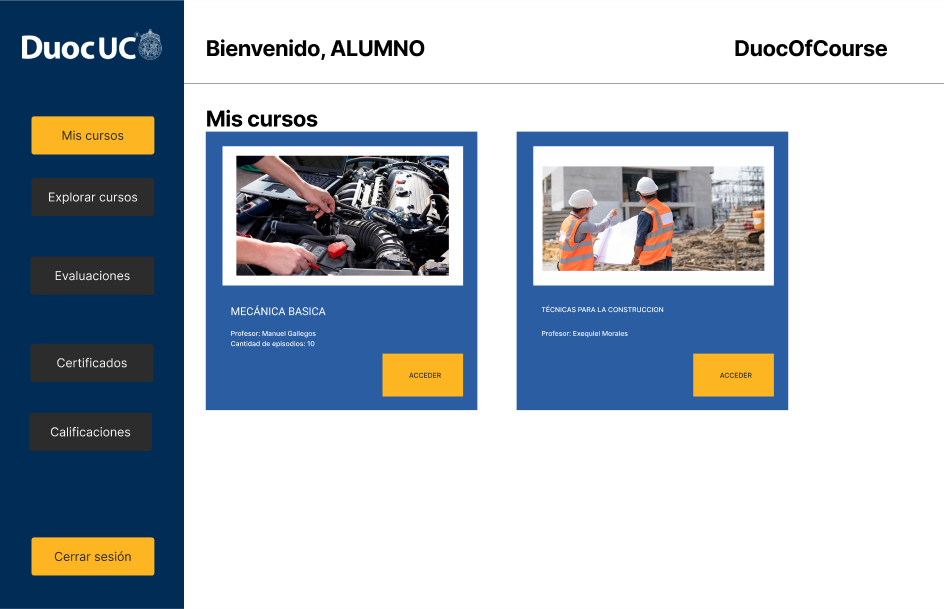
**DuocOfCourseEstudiantes.**

**Interfaz de Login.**

****

* **Descripción:** Primera interfaz que se visualiza al ingresar al sitio, se presenta un formulario para ingresar una cuenta, solicitando **correo electrónico** y **contraseña**. Además de permitir ir a la vista de registro de usuarios.
* **Propósito:** Asegurar que ninguna persona ingrese a la aplicación que no tenga una cuenta registrada previamente.
* **Definición del interfaz:** Se visualiza el nombre de la página: DuocOfCourse, debajo se presenta el formulario para ingresar el correo electrónico y la contraseña del usuario, debajo de este formulario se muestran tres botones: Iniciar sesión, Olvidé mi contraseña y Registro nuevo usuario.

**Menú principal.**

****

* **Descripción:** Menú al que se entra cuando se inicia sesión siendo **estudiante**, muestra los cursos que tiene y la navegación del estudiante.
* **Propósito:** Inicio de sesión correcto, opciones propias del **rol de estudiante**.
* **Definición del interfaz:** En este menú se visualizan los cursos que ha subido, además se muestra un botón para ver **Mis Cursos**, en el menú de la izquierda se ven las opciones de navegación del **estudiante**. En este menú se ven las siguientes opciones: **Mis cursos**, **Explorar cursos**, **Evaluaciones**, **Certificados**, **Calificaciones** , al final de este menú se muestra el botón para **Cerrar sesión**.

**Interfaz de Registro.**

****

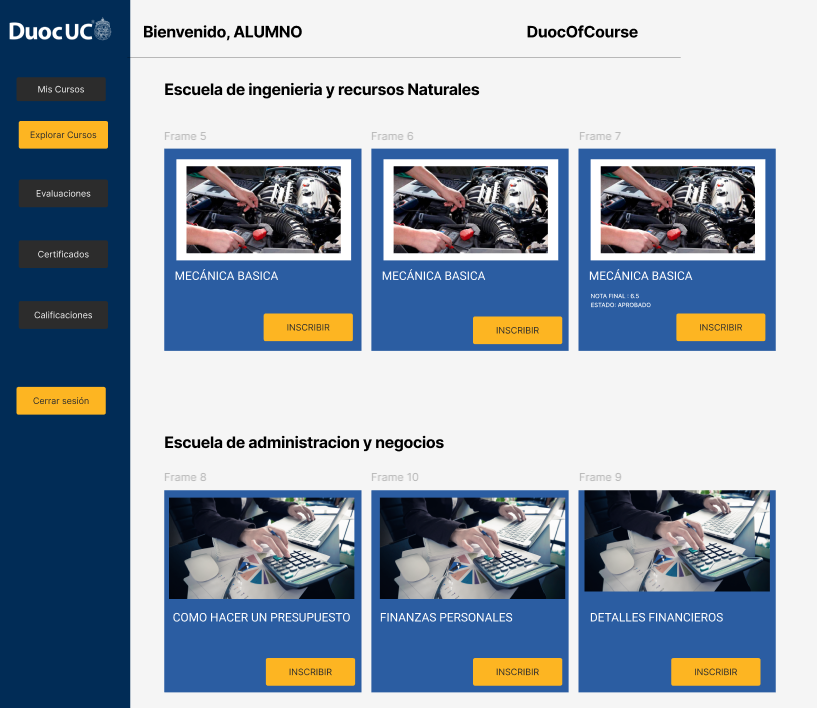
* **Descripción:** Formulario a rellenar al seleccionar el botón de **Registrarse**.
* **Propósito:** Crear una cuenta para un usuario que no tenga una cuenta previamente.
* **Definición de interfaz:** Formulario que pide el correo electrónico, nombre completo, contraseña y carrera.

**Detalle curso.**

****

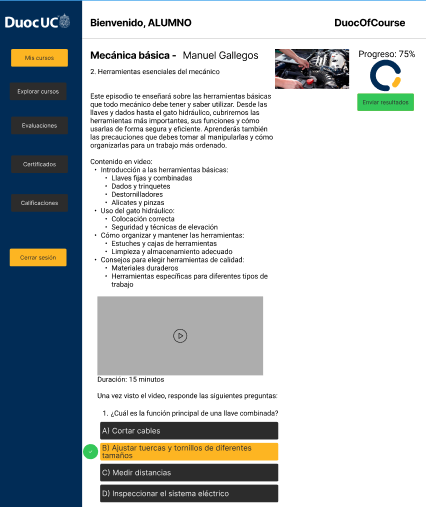
* **Descripción:** En este menú el usuario ve las clases correspondientes a un curso.
* **Propósito:** Gestionar sus cursos y realizarlos.
* **Definición del interfaz:** Se visualiza un listado de las clases pertenecientes al curso que ha accedido el estudiante, aquí se muestra el resumen de cada clase , su progreso actual en porcentaje y la calificación si lo terminó.

**Explorar cursos.**

****

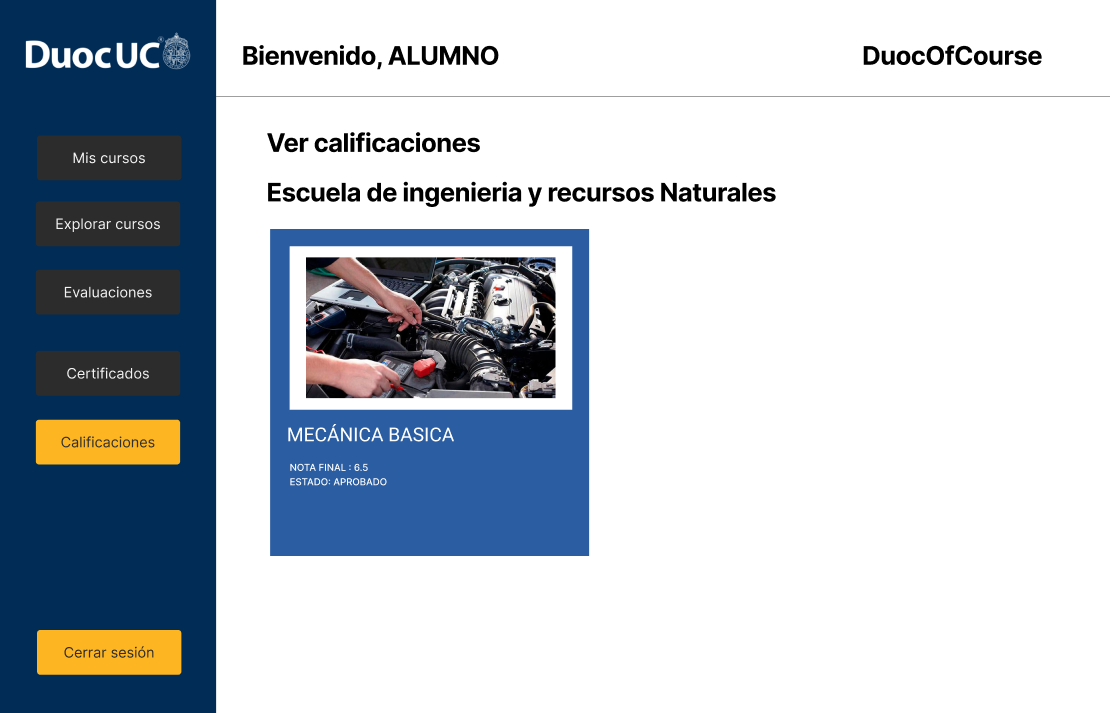
* **Descripción:** En este menú el usuario ve los cursos a los que puede inscribirse.
* **Propósito:** Buscar cursos e inscribirse.
* **Definición del interfaz:** Listado de cursos ordenados, el estudiante puede acceder a ellos mediante el botón de **Inscribir** debajo de cada curso.

**Realizar clase.**

****

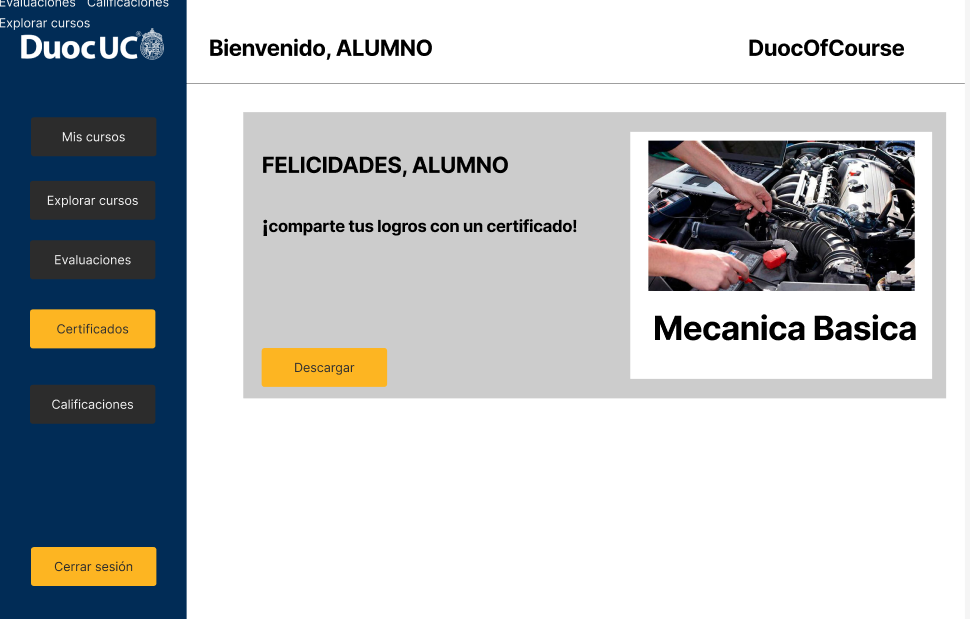
* **Descripción:** Aquí se muestra la información más detallada de la clase seleccionada por el estudiante, además del video y las preguntas si éste las tiene.
* **Propósito:** La realización del curso.
* **Definición del interfaz:** Se muestra toda la información de la clase en un apartado nuevo, lo principal es el reproductor de video que muestra la clase y las preguntas a responder debajo, el botón de **Enviar resultados** se activa cuando todas las preguntas se encuentran respondidas, luego de presionar el botón se calcula la calificación.

**Ver calificaciones.**

****

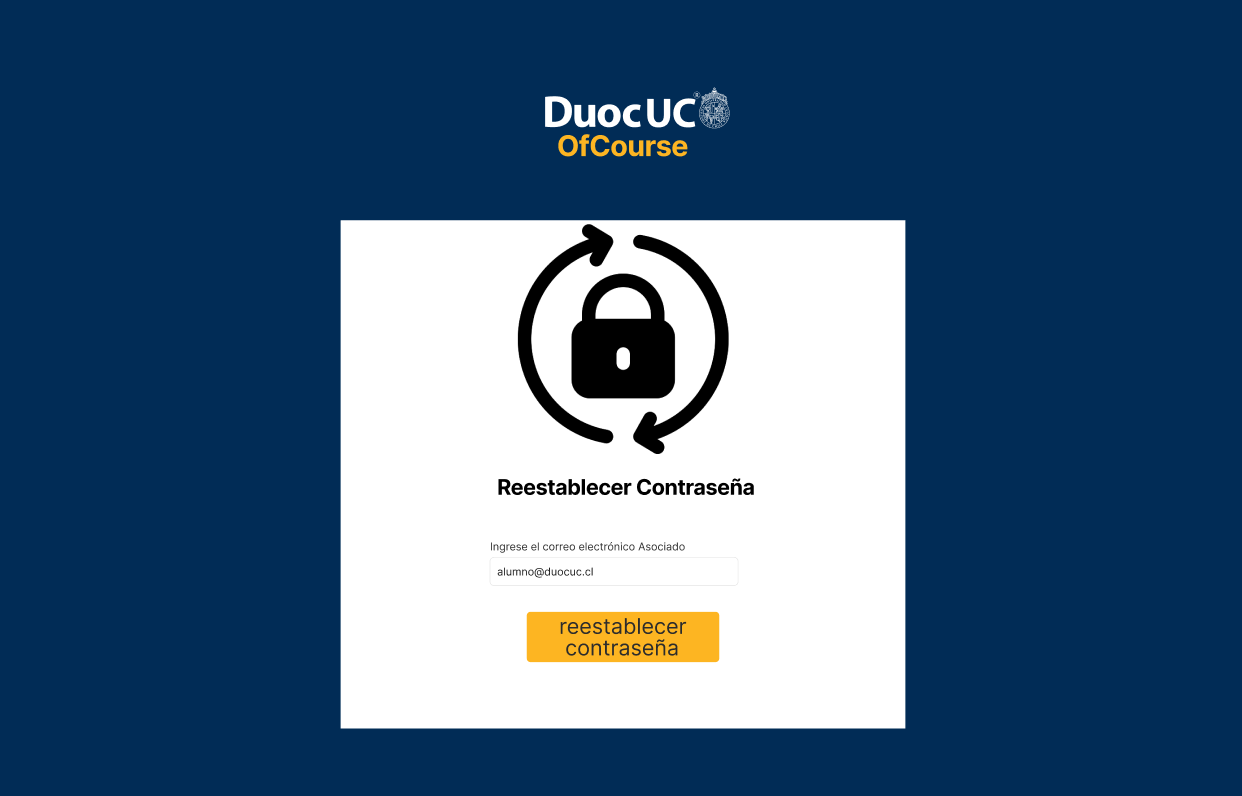
* **Descripción:** Interfaz para que los estudiantes vean sus calificaciones finales de los cursos que han realizado.
* **Propósito**:Ver las calificaciones.
* **Definición del interfaz:** Se muestra el listado de cursos realizados y sus correspondientes calificaciones, indicando si el estudiante aprobó o no el curso.

**Ver certificados.**

****

* **Descripción:** Interfaz para que los estudiantes vean sus certificados de los cursos que han aprobado.
* **Propósito**:Ver y descargar los certificados obtenidos.
* **Definición del interfaz:** Se muestra el listado de cursos aprobados, cada curso aprobado tiene un botón para **Descargar** su certificado en formato PDF.

**Recuperar contraseña.**

****

* **Descripción:** Vista que se lleva desde la interfaz de Login, aquí se puede restablecer la contraseña si el usuario la ha olvidado o si desea cambiarla.
* **Propósito:** Cambiar la contraseña.
* **Definición del interfaz:** A través del botón de Olvidé mi contraseña, se muestra este breve formulario, que muestra el campo para escribir el correo electrónico de la cuenta y un botón de Restablecer contraseña, al completarlo se redirige a un mensaje que da aviso de un e-mail que se le mandó al correo ingresado.

## 3.2 Requisitos funcionales

Los requisitos funcionales deben estar enumerados (diagrama caso uso) y pueden ser divididos en sub-secciones.

1. Iniciar sesión.
2. Crear cuenta docente.
3. Cerrar sesión
4. Gestionar cursos.
5. Subir curso.
6. Borrar curso.
7. Modificar información del curso.
8. Finalizar curso.
9. Crear exámen.
10. Evaluar calificación.
11. Gestionar cuentas de usuario.
12. Modificar usuario.
13. Generar documentos.
14. Generar informe de notas XLS.
15. Generar certificado de curso PDF.
16. Notificar por correo electrónico.
17. Recuperar contraseña.

## 3.3 Requisitos no funcionales (Organización) y de calidad (Producto)

### 3.3.1 Requisitos de Rendimiento o Eficiencia

Especificación de los requisitos relacionados con la carga que se espera tenga que soportar el sistema. Por ejemplo, el número de terminales, el número esperado de usuarios simultáneamente

**Carga y concurrencia**

* El sistema debe soportar **200 usuarios concurrentes** con degradación nula en tareas críticas (login, ver cursos, rendir evaluación).
* Debe tolerar picos de **500 RPS** (solicitudes/seg) por 5 min sin errores 5xx > **0.5%**.

**Tiempos de respuesta (P95)**

* **Login** y **cargar “Mis cursos”**: ≤ **800 ms**.
* **Acciones críticas** (inscripción, envío de evaluación): ≤ **1.5 s**.
* **Descarga de XLS/PDF**: inicio de descarga ≤ **2 s**.

**Uso de recursos (bajo carga nominal)**

* CPU backend ≤ **70%** y memoria ≤ **75%** sostenidos durante 30 min.
* MySQL (XAMPP en dev / MySQL 8 en QA/Prod): latencia media de query ≤ **100 ms**.

**Capacidad y escalabilidad**

* Debe poder **escalar horizontalmente** el backend (.NET) con balanceo, sin cambios de código.
* La base de datos debe admitir **réplicas de solo lectura** para reporting.

**Criterios de prueba**

* Pruebas de carga con planillas de 15, 30 y 60 min; la release es aceptable si se cumplen todos los P95 y tasas de error.

### 3.3.2 Requisitos de Seguridad

Especificación de elementos que protegerán al software de accesos, usos y sabotajes maliciosos, así como de modificaciones o destrucciones maliciosas o accidentales.

-Las contraseñas de las cuentas serán encriptadas y no se encontrarán en la misma tabla que los usuarios.

-En los formularios las contraseñas estarán ocultas para evitar filtraciones.

-Los usuarios tendrán permisos en base a sus roles para evitar sabotajes.

-Subidas seguras: bloquear extensiones peligrosas (.exe, .bat, .cmd, .sh, .js, .vbs, .msi, .scr) y validar MIME/tamaño (p. ej., ≤ 10 MB).

-Tráfico cifrado: habilitar HTTPS/TLS en todos los endpoints públicos; cookies HttpOnly + Secure + SameSite.

-Entradas seguras: validación/escape en backend y consultas parametrizadas a MySQL (evita inyecciones que introducen payloads).

-Mínimo privilegio: usuario de BD solo con permisos SELECT/INSERT/UPDATE/DELETE sobre su esquema; archivos accesibles solo por la app.

-Trazabilidad y respaldo: log de subidas/rechazos y backup diario con verificación de hash (SHA-256) de archivos críticos.

### 3.3.3 Requisitos de Usabilidad

**Facilidad de uso**:  
 El sistema debe ser intuitivo y fácil de aprender, incluso para usuarios con conocimientos técnicos limitados. Los usuarios deben poseer conocimientos previos de AVA Duoc y Vivo Duoc.

* **Requisito**: La interfaz de usuario debe estar diseñada de acuerdo con principios de diseño de experiencia de usuario (UX) que faciliten la navegación, con elementos visuales claros y consistentes.

**Tiempo de aprendizaje**:  
 Los usuarios deben poder realizar tareas básicas dentro del sistema en un tiempo razonable después de su primera interacción.

* **Requisito**: Los usuarios deben ser capaces de completar tareas simples (como inscribirse en un curso o revisar su progreso) en menos de 5 minutos después de iniciar sesión, sin la necesidad de instrucciones externas.

**Tiempo de respuesta**:  
 La velocidad de respuesta del sistema es crucial para garantizar una experiencia de usuario fluida. El sistema debe ser capaz de procesar solicitudes y respuestas rápidamente.

* **Requisito**: El tiempo de carga de las páginas no debe exceder los 3 segundos en conexiones estándar de Internet. Las acciones críticas (como la inscripción en cursos) deben procesarse en menos de 5 segundos.

**Manejo de errores y recuperación**:  
 El sistema debe ser capaz de manejar errores de manera que no afecten negativamente la experiencia del usuario. Los mensajes de error deben ser claros y orientados a la solución.

* **Requisito**: En caso de errores, el sistema debe proporcionar mensajes de error comprensibles, con sugerencias para que el usuario pueda corregir el problema sin necesidad de asistencia externa.
* **Requisito de recuperación**: Si el sistema sufre un fallo o desconexión, debe ser capaz de recuperar las últimas acciones del usuario sin pérdida de datos o progreso, garantizando una experiencia ininterrumpida.

### 3.3.4 Requisitos de Disponibilidad

Especificación de los factores de disponibilidad final exigidos al sistema. Normalmente expresados en % de tiempo en los que el software tiene que mostrar disponibilidad.

**Fiabilidad del sistema**:  
 El sistema debe ser confiable y robusto, asegurando que el tiempo de inactividad o los errores sean mínimos y no interrumpan el uso normal.

* **Requisito**: El sistema debe tener una disponibilidad mínima del 99.9% durante las horas pico de uso, con tiempos de inactividad planeados no superiores a 1 hora al mes.
* **Requisito de fiabilidad**: Los incidentes permitidos por el sistema no deben superar el 0.5% de los usuarios activos mensuales. En caso de fallos o errores, debe haber un procedimiento claro para la recuperación del sistema

### 3.3.5 Requisitos de Portabilidad

Identificación del tipo de mantenimiento necesario del sistema.

**Modo de ejecución para demos con VS Code Tunnels (no productivo)**

* El sistema puede exponerse temporalmente mediante **VS Code Tunnels**, publicando los puertos del **backend (.NET, 5000/5001)** y **frontend (React, 5173/3000)** como URLs HTTPS .devtunnels.ms.
* **MySQL (XAMPP, 3306)** permanece **solo local** (no se publica).
* **Requisitos**: bindings en 0.0.0.0, CORS permitiendo https://\*.devtunnels.ms y localhost, y variables de entorno para configuración (sin credenciales hardcodeadas).
* **Limitaciones**: uso exclusivo para **desarrollo/demos**, sin datos sensibles; no sustituye un entorno productivo con TLS, WAF, copias de seguridad y monitoreo dedicados.
* **Criterio de aceptación**: un revisor externo puede acceder al frontend y consumir APIs del backend vía URLs del túnel, sin cambios de código ni apertura de puertos en el router.

### 3.3.6 Requisitos de Mantenibilidad

El sistema será construido en base a microservicios, cuyas funcionalidades son independientes entre sí y por tanto si se modifica una función en particular no afectará a la aplicación.

Además constará de bloques de código los cuales estarán optimizados para ahorrar memoria y que pueden ser reutilizables dentro de la aplicación.

### 3.3.7 Requisitos de Funcionalidad

El sistema , al ser construido en base a microservicios deberá cubrir cada uno de los casos de uso que fueron propuestos por el equipo.

Además deberá satisfacer cada una de las necesidades que abarca las funcionalidades descritas, cumpliendo efectivamente cada función.

## 3.4 Requisitos No funcionales Organizacionales

1. El sistema debe parecerse en diseño y colores a la plataforma Ambiente Virtual de Aprendizaje (AVA) para facilitar la comprensión y la navegación de los usuarios.

# 4. Propuesta de Planificación

## 4.1 Descripción general acerca de la Planificación

### 4.1.2 Definición del Equipo de Trabajo

Jefe de proyecto: Encargado de la documentación principal del proyecto, organizar tareas y plazos.

Prog. Frontend / BDA: Encargado de la interfaz, prototipos y la creación y administración de la base de datos de los sistemas.

Prog. Backend: Encargado de la programación del funcionamiento de los aplicativos a desarrollar.

### 4.1.3 Definición de Actividades principales del Proyecto

Enumeres las etapas principales del proyecto de acuerdo a carta Gantt y EDT

### 4.1.4 Resumen Costos del Desarrollo del Proyecto

* Costos por FASE
  + Fase de planificación: $183.125
  + Fase de diseño: $380.000
  + Fase de desarrollo: $5.175.000
  + Fase de pruebas QA: $481.250
  + Fase de lanzamiento: $370.000
* Costos por Actor o Rol
  + Jefe de proyecto: $1.434.375
  + Prog. Frontend / BDA: $3.701.500
  + Prog. Backend: $2.047.000

### 4.1.6 Carta Gantt

Agregar las fechas más importantes definidas en la carta Gantt dejando claro los hitos o las fases principales y la fecha de término del proyecto

* Semana 6: Entrega documentación y presentación de estos.
* Semana 7: Inicio fase 2: Modelamiento de base de datos e inicio de desarrollo frontend.
* Semana 10: Inicio de desarrollo backend.
* Semana 15: Entrega del 100% de desarrollo.
* Semana 16: Inicio fase 3: Pruebas y QA.
* Semana 18: Finalización de proyecto.

## 5. Anexos

### 5.1 Acta de Proyecto

### 5.2 Matriz Especificación de Requerimientos

### 5.3 Prototipado de Software

### 5.4 Matriz EDT. Planilla Detallada Cálculo de Esfuerzo

### 5.5 Planilla Carta Gantt

Todos los documentos se encuentran adjuntados en el [[Link de Github]](https://github.com/b3njam1n18/DuocOfcourse)