

---

## **Software +Especificación de Requerimientos**

Para

Proyecto Escolar

Versión 2.0

Creada por Pablo Altamirano Arce

UNIVA

03/03/2020

## Índice

Indice.....	ii
Historial de Revision.....	ii
1. Introduccion .....	<b>Error! Bookmark not defined.</b>
1.1 Proposito .....	<b>Error! Bookmark not defined.</b>
1.2 Convenciones del Documento.....	<b>Error! Bookmark not defined.</b>
1.3 Audiencia intencionada y Sugerencias de Lectura.....	<b>Error! Bookmark not defined.</b>
1.4 Alcance del Producto .....	<b>Error! Bookmark not defined.</b>
2. Descripcion General .....	1
2.1 Perspectiva del Producto.....	1
2.2 Funciones del Producto.....	<b>Error! Bookmark not defined.</b>
2.3 Clases de Usuario y Características.....	<b>Error! Bookmark not defined.</b>
2.4 Ambiente Operativo.....	<b>Error! Bookmark not defined.</b>
2.5 Dependencias y Supuestos.....	2
3. Requerimientos de Interfaces Externas.....	2
3.1 Interfaces de Hardware .....	2
3.2 Interfaces de Software .....	2
3.3 Interfaces de Comunicación.....	<b>Error! Bookmark not defined.</b>
4. Características del Sistema .....	3
4.1 Mostrar Información Alumno .....	3

## Historial de Revisión

Nombre	Fecha	Razón de los Cambios	Versión
Pablo Altamirano Arce	03/03/20	Creación del Documento	1.0
Pablo Altamirano Arce	15/04/20	Revisión y corrección del Documento	2.0

## **1. Introducción**

### **1.1 Propósito**

Actualmente tenemos un sistema donde podemos ver tus calificaciones, las materias aprobadas y tu plan de estudios. Sin embargo hay falta de información como, porcentaje cursado de la carrera, periodos faltantes en promedio. Una información que puede ser útil para el estudiante, y así poder planear a futuro.

El proyecto planea atacar esa problemática, dando una información más útil, para poder tener una idea de cuánto falta por terminar la carrera, cuanto porcentaje lleva de la carrera y así poder planear a futuro. Cosas como, si ya estas a la mitad de la carrera tal vez sea momento de buscar prácticas profesionales o buscar un trabajo como becario, o si llevas poco de la carrera, tal vez todavía no es momento.

### **1.2 Convenciones del Documento**

Temas en negritas, texto justificado, viñetas numéricas, letra Arial 12 para letra general del documento y Arial 14, Negritas, para títulos y viñetas.

### **1.3 Audiencia intencionada y Sugerencias de Lectura**

Este documento es para el maestro de la clase de Desarrollo de Software Orientado a Objetos y el alumno desarrollador del proyecto.

### **1.4 Alcance del producto**

Es un software sencillo desarrollado en 3 capas, Front-End en React, Back-End en .NET y base de datos en SQL Server. El usuario ingresara su matrícula y esto arrojará la información actual del alumno, su carrera, promedio, el porcentaje que lleva cursado de la carrera, las materias aprobadas y un estimado de cuanto le falta para terminar. No se dará de alta información ya que el alumno no debería poder modificar su información.

## **2. Descripción General**

### **2.1 Perspectiva del Producto**

El origen de este producto es debido a la materia de Desarrollo de Software Orientado a Objetos como trabajo final. El sistema actual de la universidad no muestra información que podría ser de utilidad para el alumno, cosas como cuanto porcentaje lleva de la carrera, cuanto le falta para salir, es algo que él debe calcular manualmente, mi proyecto planea dar una información un poco más detallada, con el fin de que el alumno pueda tener una mejor información de cómo va y cuanto le falta. Se correrá de manera local, no pertenece ahorita a ningún otro sistema.

## **2.2 Funciones del Producto**

- Correrá de manera local
- Conecta a base de datos a C#
- C# API a Front
- El Usuario ingresara un número de “matricula” y la aplicación mostrara la información del alumno en pantalla.

## **2.3 Clases de Usuario y Características**

La única clase de usuario será el usuario “alumno”, no podrá modificar nada ni agregar nada, ya que un alumno no debería poder modificar sus calificaciones y sus materias. Solo podrá ingresar la matrícula y obtener la información necesaria.

## **2.4 Ambiente Operativo**

El ambiente donde funcionara el trabajo es en una computadora con Windows 10, 16gb de RAM, Intel Core i7.

El software necesario es Visual Studio 2019 Community (.NET Framework), Visual Studio Code (React/Typescript) y SQL Server, correrá de manera local.

## **2.5 Dependencias y Supuestos**

El Proyecto puede ser afectado si no se corre en el ambiente necesario, ya que no se contara con la información necesaria para poder ejecutar el código.

## **3. Requerimientos de Interfaces Externas**

### **3.1 Interfaces de Hardware**

El hardware será la laptop que tenga la información necesaria para ejecutar el código. Se debe usar teclado y mouse o trackpad para poder navegar en la interfaz.

Mínimos requerimientos:

- 4gb RAM
- Intel Core i3
- 128gbHDD

### **3.2 Interfaces de Software**

Se correrá en una laptop con Windows 10, base de datos creada en SQL Server versión 11.0, API en .NET Framework versión 4.7.2 y Front-End en React/Typescript versión 16.12.0.

### **3.3 Interfaces de Comunicación**

Se usara una API REST creada en .NET Framework, solo usando el método GET de HTTP, el archivo se recibe en formato JSON para su uso en el front.

Se usara una Servidor local creado en NODE.js donde se ejecutara el Front-End de la aplicación.

Se usara una base de datos creada en SQL Server que esta almacenada de manera local.



## **4. Características del sistema**

### **4.1 Mostrar Información Alumno**

#### **4.1.1 Descripción y prioridad**

El alumno ingresara su matrícula y esto hará que la información de su carrera se muestre en pantalla, cuantas materias ha cursado, su calificación, su promedio, el porcentaje que lleva de la carrera y un estimado de cuantos cuatrimestres le faltan para terminar la carrera. Esta es la mayor prioridad del proyecto.

#### **4.1.2 Secuencias de Estimulo/Respuesta**

El alumno deberá ingresar la matrícula en un campo de texto, luego dará clic al botón de búsqueda, y esto generara la búsqueda de su información en la base de datos, así obteniendo toda la información requerida del alumno. Una vez visible la información, el alumno puede dar clic al botón de Inicio en la pantalla, para regresar a la pantalla anterior.

#### **4.1.3 Requerimientos Funcionales**

REQ-1: Servidor Node activo desde Localhost:3000

REQ-2: Servidor API.NET Activo desde Localhost:44374

REQ-3: Base de Datos Activa en SQL Server

- REQ-4: El usuario deberá ingresar una matrícula valida en el espacio asignado en la interfaz.
- REQ-5: Se validara la matricula, primero a través de un filtro creado en el front donde se validara que la cadena conste de solamente números enteros.
- REQ-6: En caso de fallar con la validación del REQ-5 mostrara una alerta con el mensaje "matricula invalida" y la opción de volver a intentarlo.
- REQ-7: En caso de aprobar la primera validación, se pasara a una segunda validación, donde se enviara la matrícula y se hará un request GET de una función de validación en la base de datos, esta validación es para ver si existe el número de matrícula o no.
- REQ-8: En caso de que no exista se mostrara una alerta con el mensaje "matricula invalida" y la opción de volver a intentarlo.
- REQ-9: En caso de matrícula valida, se hará un request GET a la API la cual obtendrá la información.
- REQ-10: Una vez obtenida la información, se procesara y se mostrara en la interfaz gráfica, mostrando la información completa del alumno:
- Nombre
  - Edad
  - Carrera
  - Porcentaje aprobado de la carrera
  - Promedio actual
  - Periodos restantes por terminar la carrera
  - Lista de materias cursadas con su respectiva calificación
- REQ-11: Habrá un botón de "Buscar" en la barra de navegación de la interfaz.
- REQ-12: El botón "Buscar" regresara a la página inicial donde puedes ingresar la matricula del estudiante nuevamente, para hacer un nuevo request.