

---

# **Especificación de Requerimientos de Software**

**para**

## **CultAR**

**Versión 0.1 aprobada**

**Por Fernando Sosa e Ignacio Dias Gundin**

**AYDS-Project**

**08/04/2023**

# Tabla de contenidos

|   |           |
|---|-----------|
| <b>Tabla de contenidos</b>                            | <b>ii</b> |
| <b>Revisión Histórica</b>                             | <b>ii</b> |
| <b>1. Introducción</b>                                | <b>1</b>  |
| 1.1 Propósito   | 1         |
| 1.2 Conveniencias de Documentación                    | 1         |
| 1.3 Público al que Apuntamos y Sugerencias de Lectura | 1         |
| 1.4 Alcance del Producto                              | 1         |
| 1.5 Referencia  | 1         |
| <b>2. Descripción General</b>                         | <b>2</b>  |
| 2.1 Perspectiva del Producto                          | 2         |
| 2.2 Funcionalidades del Producto                      | 2         |
| 2.3 Clases y Características de los Usuarios          | 2         |
| 2.4 Entorno Operativo                                 | 2         |
| 2.5 Limitaciones de Diseño y Aplicación               | 2         |
| 2.6 Documentación del Usuario                         | 2         |
| 2.7 Suposiciones y Dependencias                       | 3         |
| <b>3. Requisitos de la Interfaz Externa</b>           | <b>3</b>  |
| 3.1 Interfaces de Usuario                             | 3         |
| 3.2 Interfaces de Hardware                            | 3         |
| 3.3 Interfaces de Software                            | 3         |
| 3.4 Interfaces de Comunicaciones                      | 3         |
| <b>4. Características de Sistema</b>                  | <b>4</b>  |
| 4.1 Características de Sistema 1                      | 4         |
| 4.2 Características de Sistema 2 (and so on)          | 4         |
| <b>5. Otros Requisitos no Funcionales</b>             | <b>4</b>  |
| 5.1 Requisitos de Rendimiento                         | 4         |
| 5.2 Requisitos de Fiabilidad                          | 5         |
| 5.3 Requisitos de Seguridad                           | 5         |
| 5.4 Atributos de Calidad de Software                  | 5         |
| 5.5 Reglas de Negocio                                 | 5         |
| <b>6. Otros Requisitos</b>                            | <b>5</b>  |
| <b>Apéndice A: Glosario</b>                           | <b>5</b>  |
| <b>Apéndice B: Análisis de Modelos</b>                | <b>5</b>  |
| <b>Apéndice C: Lista Por Determinar</b>               | <b>6</b>  |

## **Revisión Histórica**

| <b>Nombre</b> | <b>Fecha</b> | <b>Motivos de los cambios</b> | <b>Versión</b> |
|---------------|--------------|-------------------------------|----------------|
|               |              |                               |                |
|               |              |                               |                |

# 1. Introducción

## 1.1 Propósito

*El propósito del proyecto es crear una aplicación web de aprendizaje sobre Argentina, en donde se aprenderán sobre diferentes temas relacionados con el país en cuestión.*

## 1.2 Conveniencias de Documentación

*El documento fue creado a partir de la modificación del SRS\_template (creado por Karl E. Wiegers y brindado por los profesores de la materia) mediante el procesador de texto "Documentos de Google" (herramienta que forma parte de la suite Google Docs Editors). Exportado como un archivo PDF.*

## 1.3 Público al que Apuntamos y Sugerencias de Lectura

*Destinado a cualquier persona que esté interesada en aprender sobre la cultura Argentina.*

## 1.4 Alcance del Producto

*Aplicación web creada para que los usuarios aprendan sobre la cultura Argentina en diferentes ámbitos.*

## 1.5 Referencias

...

# 2. Descripción General

## 2.1 Perspectiva del Producto

*Se trata de una aplicación que busca educar al usuario sobre Argentina en los siguientes campos:*

- *Geografía*
- *Deportes*
- *Artes*
- *Historia*

## 2.2 Funcionalidades del Producto

*La función principal de la aplicación es enseñar sobre la cultura Argentina, para ello el usuario debe elegir la sección que le interese aprender y realizar las distintas preguntas en los distintos niveles de dicha sección. Una vez finalizado con un nivel se le permite realizar una evaluación*

*para validar el aprendizaje adquirido en dicho nivel y recibir la recompensa correspondiente. Tras masterizar todos los niveles de una categoría, debe hacer una evaluación final de la categoría para finalizar con ésta y recibir otra recompensa más grande.*

*Otra funcionalidad es un minijuego donde el usuario debe responder lo más rápido posible, a contra reloj, preguntas al azar de distintos temas.*

*También el usuario tiene acceso a una biblioteca en donde encontrará varios textos relacionados con preguntas que se hacen en los juegos y/o evaluaciones.*

## **2.3 Clases y Características de los Usuarios**

## **2.4 Entorno Operativo**

*Desarrollado utilizando HTML, CSS y Ruby ...*

## **2.5 Limitaciones de Diseño y Aplicación**

*La aplicación debe ser rápida y sencilla facilitando el aprendizaje de los usuarios. Toda la información y los juegos son exclusivamente sobre la cultura Argentina y no necesariamente hay relación con la cultura de otros países.*

## **2.6 Documentación del Usuario**

*El usuario podrá encontrar un manual ubicado dentro de la opción de configuración de aplicaciones. Este manual describe las funciones de la aplicación y cómo utilizarlas.*

## **2.7 Suposiciones y Dependencias**

*Debe asegurarse de que toda la información es correcta.*

*La aplicación debe satisfacer las necesidades del cliente y se pueden añadir o cambiar características si se necesita.*

# **3. Requisitos de la Interfaz Externa**

## **3.1 Interfaces de Usuario**

*La GUI (Interfaz Gráfica del Usuario, IGU) buscará ser simple visualmente para que el usuario pueda encontrar lo que busca de manera rápida.*

### 3.2 Interfaces de Hardware

*Al ser una aplicación web, se necesitará de un navegador web para poder utilizarla. Por lo que se necesitará de una computadora o dispositivo móvil que cuente con un navegador.*

### 3.3 Interfaces de Software

*Usamos Ruby, HTML y CSS.*

### 3.4 Interfaces de Comunicaciones

*<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.>*

## 4. Características de Sistema

### 4.1 Validación de Usuario

#### 4.1.1 Descripción y Prioridad

*La creación y validación de un usuario es sumamente importante ya que necesitamos guardar todo el progreso para cada uno de los usuarios y también toda su información de su cuenta.*

*La validación/creación de una cuenta es utilizando un correo electrónico y una contraseña.*

#### 4.1.2 Secuencia Estímulo/Respuesta

*El usuario solo debe ingresar su correo electrónico y su respectiva contraseña, para iniciar la sesión en su cuenta.*

#### 4.1.3 Requisitos Funcionales

*Para realizar esto necesitamos una base de datos que almacene los correos electrónicos y contraseña de cada usuario para poder validar sus cuentas.*

REQ-1: Debe ser simple

### 4.2 Sistema de Juego

#### 4.2.1 Descripción y Prioridad

*Esta función es una de las principales, ya que permite comenzar con el juego de una categoría, al seleccionarla va a poder responder las preguntas que están en cada nivel. Cada categoría está subdividida en niveles que tienen preguntas propias de dicha categoría.*

#### 4.2.2 Secuencia Estímulo/Respuesta

*El usuario debe seleccionar la categoría que desee para poder entrar al menú correspondiente, en dicho menú cuenta con una ambientación adecuada para la categoría que eligió y verá los diversos niveles que debe completar para masterizar la categoría. Dentro de cada nivel le saldrán diversas preguntas adecuadas a la dificultad del nivel. Las preguntas cuando comienza el juego no tienen tiempo límite para ser respondidas*

#### 4.2.3 Requisitos Funcionales

*Se necesitará una base de datos que contenga todas las preguntas de cada nivel y sus respectivas respuestas para realizar la corrección a los usuarios. También es necesario almacenar la cantidad de preguntas correctas de un usuario. Se le aumenta el puntaje al usuario por cada respuesta correcta.*

REQ-1: La corrección de respuestas debe ser veloz

REQ-2: Por cada respuesta errónea se debe mostrar una ayuda

### 4.3 Sistema de Evaluación

#### 4.3.1 Descripción y Prioridad

*Esta función es de alta prioridad ya que nos permite ir validando el aprendizaje de cada usuario al finalizar cada nivel de una categoría específica. Consta de varias preguntas que derivan de las realizadas en el nivel.*

#### 4.3.2 Secuencia Estímulo/Respuesta

*Las evaluaciones se encuentran dentro de cada nivel perteneciente a cada categoría. El usuario puede realizarlas al tener respondidas correctamente cada pregunta del nivel, al comenzar con una evaluación el usuario dispone de 30 seg para responder cada pregunta correctamente. Para aprobar la evaluación debe tener el 100% de las preguntas correctas.*

#### 4.3.3 Requisitos Funcionales

*Se necesitará una base de datos que almacene cada pregunta que tenga cada evaluación y sus respectivas respuestas correctas. Se le aumenta el puntaje al usuario de acuerdo a la dificultad del examen*

REQ-1: La corrección de respuestas debe ser veloz

REQ-2: Si desaprueba la evaluación se le debe mostrar una ayuda

## 4.4 Sistema de Premios

### 4.4.1 Descripción y Prioridad

*Esta función es secundaria, Se van otorgando premios al usuarios al cumplir determinados logros, estos premios son mejoras consumibles que les brinda cierta ayuda/ventaja a la hora de responder preguntas.*

### 4.4.2 Secuencia Estímulo/Respuesta

*Los usuarios premiados al haber conseguido un logro, sólo podrán usar las mejoras en preguntas que se encuentran dentro de los niveles de cada categoría. No se puede usar mejoras en evaluaciones o en eventos especiales.*

### 4.4.3 Requisitos Funcionales

*Se debe mantener en una base de datos la información sobre el tipo de mejoras que hay y los logros que se deben cumplir para dar los premios, también es necesario disponer los avances de cada usuario para saber si se cumplió algún logro.*

REQ-1: El otorgamiento de premios debe ser veloz

## 4.5 Sistema de Ranking

### 4.5.1 Descripción y Prioridad

*El ranking es una funcionalidad secundaria, se basa en el puntaje obtenido al responder preguntas, realizar evaluaciones y jugar eventos especiales. Estos puntajes se exhiben en una tabla junto al nombre del usuario que le corresponde.*

### 4.5.2 Secuencia Estímulo/Respuesta

*El usuario va ganando puntaje a medida que responde preguntas correctamente, completa evaluaciones y juega eventos especiales, lo que le permite ir subiendo de ranking y a su vez dependiendo de su posición en el ranking va recibiendo insignias que lo caracterizan.*

### 4.5.3 Requisitos Funcionales

Se debe almacenar en una base de datos la posición en el ranking de cada usuario junto con su puntaje y las insignias que posee. A su vez también es necesario almacenar qué puntaje/posición es necesaria para otorgarle una insignia determinada a cada usuario.

REQ-1: El otorgamiento de Insignias debe ser veloz

REQ-2: La actualización del ranking debe ser veloz y continua



## 4.6 Sistema de Eventos Especiales

### 4.6.1 Descripción y Prioridad

*Los eventos especiales es una funcionalidad extra secundaria, consta de actualmente de un minijuego en el que se debe responder el mayor número de preguntas a contra reloj.*

### 4.6.2 Secuencia Estímulo/Respuesta

*Los usuarios que deseen jugar este minijuego tienen que responder el mayor número de preguntas al azar, de distintas categorías, teniendo un tiempo total de 40 segundos. Las preguntas se pueden responder o saltar, si una pregunta se responde erróneamente termina el juego sin importar el tiempo sobrante.*

### 4.6.3 Requisitos Funcionales

*Se necesita almacenar las preguntas que se van a utilizar en el juego y sus respectivas respuestas correctas, además se debe almacenar la cantidad de preguntas correctas que respondió el usuario.*

*Se aumenta el puntaje del usuario dependiendo de la cantidad de preguntas correctas.*

REQ-1: El paso de pregunta a pregunta debe ser veloz

REQ-2: La corrección de respuestas debe ser veloz

## 4.7 Sistema de Biblioteca

### 4.7.1 Descripción y Prioridad

*La biblioteca es una funcionalidad útil pero secundaria, contiene textos informativos sobre temas y preguntas de cada categoría y nivel.*

### 4.7.2 Secuencia Estímulo/Respuesta

*El usuario puede acceder a la biblioteca y utilizar todos los textos que se encuentren allí*

### 4.7.3 Requisitos Funcionales

*Se va a necesitar almacenar en una base de datos todos los textos que se encuentren en la biblioteca, y a la categoría que pertenezcan.*

## 4.8 Sistema de FeedBack

### 4.8.1 Descripción y Prioridad

*Esta funcionalidad es secundaria, le permite al usuario enviar reportes sobre problemas en el juego o mejoras que podríamos aplicar.*

### 4.8.2 Secuencia Estímulo/Respuesta

*Le permite al usuario enviar un mensaje a los autores de la aplicación, sobre algún error que hayan experimentado o sobre mejoras que les gustaría que contenga el juego.*

#### 4.8.3 Requisitos Funcionales

*En una base de datos se va a necesitar almacenar cada FeedBack de cada usuario*

## 5. Otros Requisitos no Funcionales

### 5.1 Requisitos de Rendimiento

*<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>*

### 5.2 Requisitos de Fiabilidad

*La aplicación se puede usar en celulares o portátiles, por lo que el usuario debe ser consciente de los alrededores al usar la aplicación.*

### 5.3 Requisitos de Seguridad

*El usuario ingresará un nombre para identificarse, una foto de perfil y su descripción. Si bien recomendamos que no ingresen información privada, debemos protegerla en caso de que decidan hacerlo.*

### 5.4 Atributos de Calidad de Software

*Se diseñará para navegadores web utilizados desde una PC. Más adelante se verá el uso desde un dispositivo móvil (en estos últimos se podrá entrar y funcionará correctamente, pero visualmente no será cómodo para el usuario ya que no está pensado para sus dimensiones)*

### 5.5 Reglas de Negocio

*<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>*

## 6. Otros Requisitos

*<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>*

### Apéndice A: Glosario

*<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>*

### Apéndice B: Análisis de Modelos

*<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>*

### Apéndice C: Lista Por Determinar

*<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>*

