Tabla de contenido

[1 Generalidades del proyecto 3](#_Toc76375133)

[1.1 Nombre y Especificaciones Generales: 3](#_Toc76375134)

[2 Documentación 3](#_Toc76375135)

[2.1.1 Casos de Uso 3](#_Toc76375136)

[2.2 Historias de Usuario 4](#_Toc76375137)

[3 Mecánica de Coordinación 4](#_Toc76375138)

[3.1 Roles 5](#_Toc76375139)

[3.1.1 Backend 5](#_Toc76375140)

[3.1.2 Frontend 5](#_Toc76375141)

[3.1.3 Bases de Datos 5](#_Toc76375142)

[3.2 Gestor del Proyecto 5](#_Toc76375143)

[4 Pruebas 5](#_Toc76375144)

[5 Interfaz Grafica 5](#_Toc76375145)

[5.1 Especificaciones 5](#_Toc76375146)

[5.2 Funcionalidad 5](#_Toc76375147)

[6 Pruebas 5](#_Toc76375148)

[7 Lenguajes de Programación y *Stack* Tecnológico 5](#_Toc76375149)

[7.1 Lenguajes de Programación: 5](#_Toc76375150)

[7.2 Java Script 5](#_Toc76375151)

[7.3 *Stack* Tecnológico 5](#_Toc76375152)

[7.3.1 Postman: 5](#_Toc76375153)

[Se utilizará para realizar el testing, consumir y depurar API REST. Además, para probar todas las peticiones del servidor. 5](#_Toc76375154)

[7.3.2 Visual Studio Code: 5](#_Toc76375155)

[7.3.3 Node JS versión en LTS: 6](#_Toc76375156)

[7.3.4 YARN 6](#_Toc76375157)

[7.3.5 MongoDB 7](#_Toc76375158)

[7.3.6 Robo 3T 7](#_Toc76375159)

[8 Bases de Datos y APIs 7](#_Toc76375160)

[9 Repositorios 7](#_Toc76375161)

[10 Plataforma Tecnológica en la Nube 7](#_Toc76375162)

[10.1 Acceso y Uso 7](#_Toc76375163)

[11 Ambientes de Desarrollo y Producción 7](#_Toc76375164)

[11.1 Configuración 7](#_Toc76375165)

[11.2 Uso 7](#_Toc76375166)

# Generalidades del proyecto

## Nombre y Especificaciones Generales:

GestorProf, es una página web diseñada para que el docente que la adquiera pueda subir documentación de sus clases y esta se pueda visualizar por sus estudiantes, estos documentos pueden ser de texto, imágenes o videos que previamente estar dispuestos en la base de datos.

El docente será el administrador de la plataforma, y los estudiantes sus usuarios principales.

# Documentación

### Casos de Uso

|  |  |  |  |
| --- | --- | --- | --- |
| Formato Casos de Uso | | | V-1 |
| **Nombre del caso de uso:** | | **CU.03 Edicion de Articulos** | |
| Descripcion del Caso de Uso | | | |
| **Actores:** | | Act 1. Administrador | |
| **Descripcion:** | | El administrador edita un articulo | |
| **Disparador:** | | Click en el boton de editar | |
| **Precondiciones:** | | Usuario y contraseña son de tipo administrador | |
| **Pos Condiciones:** | | Modificacion del articulo | |
| Flujo Normal | | | |
| **N°** | **Actor** | **N°** | **Acción** |
|  |  | **1** | El sistestema muestra pantallana de backend |
| **2** | El administrador clickea en articulo |  |  |
| **3** | El administrador busca el articulo a editar |  |  |
| **4** | El administrador hace las modificaciones |  |  |
| **5** | Da click en aceptar |  |  |
|  |  | **6** | El sistema modifica el articulo |
| Flujos Alternativos | | | |
| **N°** | **Actor** | **N°** | **Acción** |
| **\_** | El administrador puede cancelar la operación en cualquier momento. Finaliza caso de uso |  |  |
|  |  |  |  |

## Historias de Usuario

|  |  |
| --- | --- |
| Formato Historias de Usuario V-1 | |
| **ID:** | 1 |
| **Nombre:** | Registrar usuario administrador |
| **Prioridad del negocio:** | Alta |
| **Iteración asignada:** |  |
| **Rol:** | Yo como usuario root |
| **Funcionalidad:** | Necesito crear usuario administrador |
| **Criterio de aceptación:** | \* El usuario debe proporcionar e-mail \* El usuario debe proporcionar contraseña |

# Mecánica de Coordinación

Se realizarán reuniones diarias para determinar las tareas a realizar y el responsable de estas dependiendo del rol previamente establecido, en estas reuniones también se evaluaran las actividades que ya se han hecho y si hay posibilidades de mejora o ampliación.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **COMPROMISO** | **RESPONSABLE** | **ROL** | **FECHA DE**  **ASIGNACIÓN** | **FECHA DE ENTREGA** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## Roles

### Backend

### Frontend

### Bases de Datos

## Gestor del Proyecto

# Pruebas

# Interfaz Grafica

## Especificaciones

## Funcionalidad

# Pruebas

# Lenguajes de Programación y *Stack* Tecnológico

## Lenguajes de Programación:

## Java Script

## *Stack* Tecnológico

### Postman:

### Se utilizará para realizar el testing, consumir y depurar API REST. Además, para probar todas las peticiones del servidor.

### Visual Studio Code:

Es el editor de código a utilizar, es compatible con el lenguaje de programación a utilizares desarrollado por Microsoft además de ser totalmente gratuito.

Las extensiones a utilizar son:

* Bracket pair colorizer
* ES7 React/Redux/GraphQL/React-Native snippets: snippers para reack
* ESLint: corregir errores de sintaxis en java script
* HTML Snippets: completa código HTML
* Intellisense for CSS class names in HTML: autocompletar o auto rellenar las clases CSS
* JavaScript (ES6) code snippets: autorellenar
* Material Icon Theme: Iconos para todos los archivos
* Path intellisense: completa un path a la hora de integrarlo
* Prettier –Code formatter: Indenta código
* Reactjs code snippets: Disparadores de código React JS
* Vscode-pdf: Lector de PDF

### Node JS versión en LTS:

Es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación JavaScript, asíncrono, con E/S de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google. Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables, como, por ejemplo, servidores web.4.

### YARN

Es un nuevo tipo de instalador de paquetes JavaScript y gestor de dependencias lanzado por la empresa Facebook en colaboración con otros desarrolladores como Google donde introduce cambios en esa gestión de dependencias, en la ejecución de tareas y algunas mejoras de rendimiento, también en el cambio de enfoque en la descarga e instalación de los paquetes y en su gestión de las dependencias, por ejemplo, con Yarn el programador podrá gestionar nuestras dependencias con mayor fiabilidad.1Es compatible con el registro del npm, pero difiere en su acercamiento a instalar paquetes; Utiliza archivos de bloqueo y un algoritmo de instalación determinista, esto le permite mantener la misma estructura de los directorios node modules que albergan dependencias-para todos los usuarios involucrados en un proyecto, y ayudar a reducir los errores que son difíciles derastrear y replicar en múltiples máquinas.

### MongoDB

Es un sistema de base de datos NoSQL, orientado a documentos y de código abierto. En lugar de guardar los datos en tablas, tal y como se hace en las bases de datos relacionales, MongoDB guarda estructuras de datos BSON (una especificación similar a JSON) con un esquema dinámico, haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida. MongoDB es una base de datos adecuada para su uso en producción y con múltiples funcionalidades. Esta base de datos se utiliza mucho en la industria, contando con implantaciones en empresas como MTV Network, Craiglist, Foursquare.

### Robo 3T

Es una GUI (Gestor de BD) ligera gratuita para MongoDB. Su principal objetivo es ejecutar consultas, crear índices y visualizar documentos.

# Bases de Datos y APIs

# Repositorios

# Plataforma Tecnológica en la Nube

## Acceso y Uso

# Ambientes de Desarrollo y Producción

## Azure DevOps

Azure DevOps proporciona servicios para desarrolladores para apoyar a los equipos de soporte técnico a planear el trabajo, colaborar en el desarrollo de código, y compilar e implementar aplicaciones. Los desarrolladores pueden trabajar en la nube con *Azure DevOps Services*. (Microsoft, 2021)

### Configuración

### Uso