

# **Sistema de Gestión de Producción Científica del IFSul Campus Santana do Livramento**

**Emerson Rodrigo Prieto Da Silva<sup>1</sup>, Alfredo Parteli Gomes<sup>2</sup>, Abner Gilead Araujo Guedes<sup>3</sup>**

<sup>1</sup> Alumno del curso de Tecnologías en Análisis y Desarrollo de Sistemas. Instituto Federal Sul-rio-grandense Campus Sant'ana do Livramento (IFSul)

<sup>2</sup> Profesor Orientador. Instituto Federal Sul-rio-grandense Campus Sant'ana do Livramento (IFSul)

<sup>3</sup> Profesor Co-orientador. Instituto Federal Sul-rio-grandense Campus Sant'ana do Livramento (IFSul)

rodrigoprietods@gmail.com, alfredogomes@ifsul.edu.br,  
abnerguedes@ifsul.edu.br

## **1. Introducción**

En el contexto actual, la gestión eficiente de la producción científica es un elemento fundamental para las instituciones educativas y de investigación. El Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense (IFSul), Campus Santana do Livramento, enfrenta un desafío crucial: la falta de un catálogo que recoja y organice los trabajos de conclusión de curso (TCC), proyectos de investigación de pregrado y otros trabajos académicos producidos en sus instalaciones (Da Silva, 2023). Esta carencia representa un obstáculo significativo en la difusión y el acceso a la producción intelectual generada por la comunidad académica del campus (De Amorim, 2015).

En respuesta a esta necesidad, se propone el desarrollo del proyecto "Sistema de Gestión de Producción Científica del IFSul Campus Santana do Livramento". Este proyecto busca abordar esta carencia mediante la creación de una plataforma web que permita la catalogación, búsqueda y seguimiento de la producción científica, así como la facilitación del proceso de selección de temas de TCC y orientadores.

### **1.1. Objetivo general**

Desarrollar un sistema de gestión de producción científica para el IFSul Campus Santana do Livramento que permita la catalogación, búsqueda y seguimiento de trabajos de conclusión de curso (TCC), proyectos e investigaciones de todos los niveles académicos, además de facilitar la selección de temas de TCC y orientadores.

### **1.2. Objetivos específicos**

- Crear un catálogo público de producción científica con opciones de búsqueda por título, área, periodo, orientador, autor, curso, tipo de producción y palabras clave.
- Permitir la visualización detallada de cada producción científica, incluyendo información como tema, título, área, resumen, PDF del trabajo, palabras clave, autor, orientador, co-orientador (si aplica) y publicaciones.

- Implementar una base de datos de ideas de temas de TCC y demandas para que los estudiantes puedan seleccionar o proponer temas.
- Generar términos de compromiso en PDF para formalizar la selección de temas y orientadores.
- Habilitar la visualización de la herramienta desarrollada o su código fuente a través de la red privada de IFSul.

### **1.3. Justificación**

El desarrollo de este sistema de gestión de producción científica es esencial para el IFSul Campus Santana do Livramento por varias razones.

En primer lugar, busca valorizar y mejorar la visibilidad y accesibilidad de la producción científica de la institución. Esta mejora contribuirá a incentivar y facilitar la búsqueda y consulta de trabajos académicos, brindando a estudiantes, docentes y la comunidad en general un acceso más eficiente y completo a este contenido.

Además, el sistema tiene como objetivo agilizar el proceso de selección de temas de Trabajo de Conclusión de Curso (TCC) y orientadores, simplificando así la toma de decisiones para los estudiantes. Al mismo tiempo, se propone como una herramienta para promover la colaboración con otras instituciones, facilitando el acceso a repositorios externos y fomentando el intercambio de conocimientos y recursos.

Finalmente, el sistema se erige como un medio para ofrecer mecanismos efectivos de gestión de usuarios y contenido, consolidando una plataforma que atiende las necesidades de la comunidad académica y fortalece la posición del IFSul Campus Santana do Livramento en el ámbito científico y educativo.

## **2. Marco teórico y tecnologías a utilizar**

En esta sección se presenta el marco teórico, incluyendo los trabajos relacionados y las tecnologías que se utilizarán para el desarrollo del sistema.

### **2.1. Marco teórico**

Según Marques (2019), el Trabajo de Conclusión de Curso (TCC) juega un papel crucial al involucrar a los estudiantes en el contexto de la investigación. Esto los lleva a buscar nuevos conocimientos, explorar temas, discutir y generar nuevas ideas como parte de su jornada para resolver problemas específicos. Esta perspectiva es similar a la de Severino (2007, citado en Silva, 2023), quien enfatiza que el TCC proporciona una experiencia valiosa en términos de aprendizaje y adquisición de conocimientos.

Según el Manual de Trabajo de Conclusión de Curso de la Faculdade Santíssima Trindade (2022), el TCC es la elaboración de documentos estructurados que mejoran la comprensión de conceptos previamente aprendidos durante la graduación. El manual enfatiza que la investigación científica es una parte fundamental del TCC. Describe el TCC como una tarea que implica una investigación profunda y que a menudo va más allá de la escritura, abarcando el análisis crítico y la reflexión analítica. La planificación académica del TCC, tal como se describe en el manual, tiene como objetivo mejorar las habilidades de los estudiantes en lectura, análisis, crítica y discusión de contenidos, a

menudo en forma de artículo científico, de acuerdo con los estándares de la ABNT (Faculdade Santíssima Trindade, 2022).

Es importante señalar que en el Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense (IFSul) Campus Santana do Livramento, donde está inscripto el autor de esta investigación, las directrices básicas para el TCC están definidas en el Proyecto Pedagógico del Curso (PPC). Según el PPC (2023), el TCC se considera un requisito para la obtención de la certificación final y la emisión del diploma. Esta situación plantea importantes interrogantes sobre la relevancia de establecer una colección digital de proyectos de investigación en el IFSul en el campus en cuestión, con el fin de facilitar el desarrollo efectivo del TCC.

Según el estudio de Freitas (2012), entre las diversas materias que enfrentan los estudiantes en los cursos de pregrado, el TCC es generalmente visto como un desafío y, en ocasiones, considerado un "obstáculo". Además, es fundamental que los estudiantes se sientan realmente interesados en el tema elegido y confiados en su elección para evitar desviaciones en el proceso de investigación, ya que el TCC normalmente se realiza en la etapa final de graduación (Pereira & Silva, 2011).

Según Lenoir (2006), realizar toda la investigación y el trabajo que requieren las diferentes disciplinas para preparar el TCC requiere una dedicación considerable por parte de los estudiantes en términos de tiempo, siendo este uno de los requisitos esenciales para su ejecución.

Desde la perspectiva de Murakami y Fausto (2013), los repositorios digitales juegan un papel crucial en la rápida evolución de la comunicación científica en el entorno virtual. Viana y Arellano (2006) definen los repositorios digitales como plataformas de almacenamiento de contenidos digitales con la capacidad de gestionar grandes volúmenes de material durante largos períodos de tiempo.

Según Almeida, Oliveira y Rosa (2019), los repositorios digitales institucionales (RDIs) surgieron como respuesta a la creciente industrialización de la comunicación científica. Desempeñan un papel fundamental en la preservación y difusión de la memoria institucional de las universidades, integrándose cada vez más en las actividades académicas cotidianas. Almeida, Oliveira y Rosa (2019) destacan que las RDIs se han popularizado en el ámbito académico debido a su capacidad de brindar acceso rápido a las producciones científicas. Destacan la importancia de estos repositorios como herramientas de acceso abierto que promueven la organización, difusión y preservación de la memoria institucional. Para los autores, las RDIs desempeñan un papel vital en la misión de la universidad de ser un centro de producción intelectual, contribuyendo al avance de la ciencia, la tecnología y la humanidad.

En este sentido, la creación de un sistema de gestión de producción científica en el IFSul Santana do Livramento no solo serviría como un medio para difundir la investigación, sino también como un medio para preservar la memoria institucional y proporcionar ejemplos de proyectos de investigación ya realizados por los estudiantes (Silva, 2023).

## 2.2. Trabajos relacionados

Se encontraron 6 artículos de trabajos que se relacionaban directamente con el proyecto. Si bien esos trabajos relacionados ofrecen la catalogación de producciones científicas al igual que este proyecto, se constató que ninguno de ellos ofrece la posibilidad de seleccionar el tema de TCC y el orientador. En la Tabla 1 se muestra una comparación entre este proyecto y los trabajos relacionados.

**Tabla 1. Tabla comparativa entre este proyecto y otros trabajos relacionados**

<b>Trabajos relacionados</b>	<b>Metodología</b>	<b>Lenguaje o herramienta</b>	<b>Base de datos</b>	<b>Web / Móvil</b>	<b>Seleccionar tema de TCC y orientador</b>
Este proyecto	Cascada	Java	MySQL	Web	Sí
Da Silva & Silva (2022)	Prototipado	JavaScript	No especificado	Móvil	No
Pavão et al. (2008)	No especificado	DSpace	PostgreSQL	Web	No
Palomino et al. (2010)	No especificado	Lucene	No especificado	Web	No
Rodrigues et al. (2004)	No especificado	DSpace	PostgreSQL	Web	No
Ramírez et al. (2019)	SCRUM	CMS	No especificado	Web	No
De Amorim (2015)	Cascada	PHP	MySQL	Web	No

## 2.3. Tecnologías a utilizar

Este sistema se desarrollará utilizando las siguientes tecnologías:

### 2.3.1. MySQL

MySQL es un sistema de gestión de bases de datos SQL de código abierto muy popular, desarrollado y respaldado por Oracle Corporation. Este sistema se utiliza para administrar bases de datos relacionales y sigue el estándar SQL. Una característica importante es su naturaleza de código abierto, lo que permite a los usuarios descargarlo, estudiar su código fuente y modificarlo según sus necesidades. MySQL es conocido por ser rápido, confiable y escalable, adecuado para una variedad de aplicaciones, desde aplicaciones de escritorio hasta entornos de producción en red. Además, ofrece una amplia gama de herramientas y características, incluyendo soporte multilingüe y

múltiples juegos de caracteres. El sistema también se puede utilizar en entornos de cliente/servidor o como una biblioteca incrustada en aplicaciones (MySQL, 2023).

La historia de MySQL se remonta a la decisión de los fundadores de crear una interfaz SQL rápida y flexible para sus propias necesidades, lo que resultó en MySQL. El nombre de MySQL proviene de la hija de uno de los cofundadores, Monty Widenius, y su logotipo es un delfín llamado "Sakila," elegido a través de un concurso de nombre entre los usuarios (MySQL, 2023).

La siguiente tabla (ver Tabla 2) muestra una comparación detallada entre tres sistemas de gestión de base de datos relacionales: MySQL, PostgreSQL y SQLite (LovTechnology, 2023)

**Tabla 2. Comparación entre MySQL y otros sistemas de gestión de base de datos relacionales**

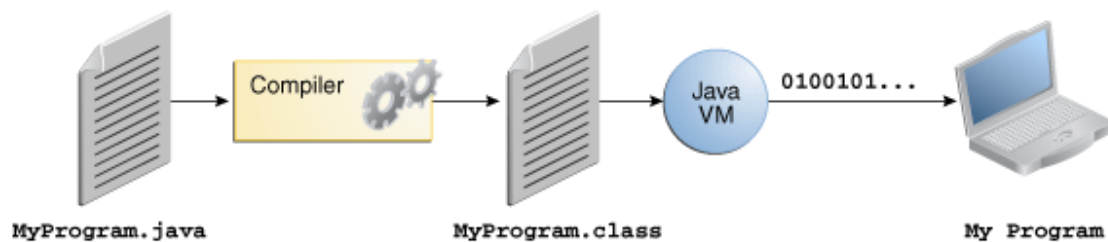
<b>MySQL</b>	<b>PostgreSQL</b>	<b>SQLite</b>
Es de uso libre y gratuito	Es de código libre y gratuito	Es de código abierto
Presenta una excelente seguridad y encriptación	Es muy estable y proporciona mucha confiabilidad	No presenta muchas funciones de seguridad y administración de usuarios
Presenta una buena velocidad al ejecutar las operaciones	Es lento en el caso de las inserciones y actualizaciones en bases de datos pequeñas	Da la posibilidad de operaciones de lectura y escritura más rápidas.
Es fácil de instalar y configurar	Tiene una herramienta gráfica con la finalidad de administrar las bases de datos de manera fácil e intuitiva	Es muy fácil de usar, se puede configurar incluso en teléfonos móviles, y no es necesario gastar mucho esfuerzo para configurarlo.
Es multiplataforma	Es multiplataforma	Soporta una amplia gama de plataformas
No consume demasiados recursos informáticos	Consume más recursos que MySQL	Usa procesos eficientes para almacenar datos
No es muy eficiente al trabajar bases de datos demasiados grandes	Tiene una gran capacidad de almacenamiento	Es muy difícil escalar en esta base de datos

**2.3.2. Java**

Java es un lenguaje de programación y plataforma informática que se lanzó por primera vez en 1995. Ha evolucionado desde sus humildes comienzos para impulsar una gran parte del mundo digital actual, proporcionando una plataforma confiable en la que se

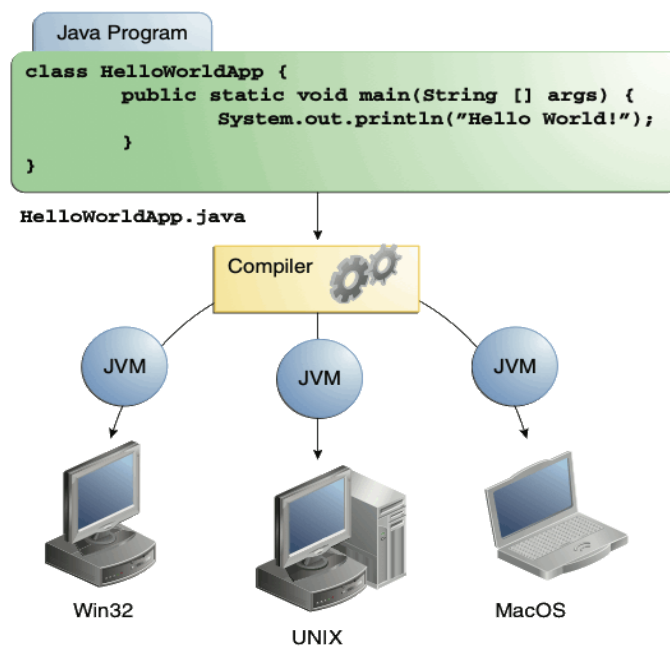
construyen muchos servicios y aplicaciones. Aunque la mayoría de las aplicaciones modernas de Java combinan el tiempo de ejecución de Java y la aplicación en un solo paquete, todavía existen muchas aplicaciones y sitios web que no funcionarán a menos que tengas Java instalado en tu escritorio (Java, 2023).

El lenguaje de programación Java se caracteriza por ser simple, orientado a objetos, distribuido, multihilo, dinámico, neutral en cuanto a la arquitectura, portátil, de alto rendimiento, robusto y seguro. En Java, todo el código fuente se escribe en archivos de texto plano con extensión .java, que luego se compilan en archivos .class con el compilador javac (Figura 1). Estos archivos .class contienen bytecodes, que son el lenguaje de máquina de la Máquina Virtual de Java (Java VM), y la herramienta de lanzamiento java ejecuta la aplicación con una instancia de la Máquina Virtual de Java (Oracle, 2023).



**Figura 1.** Una visión general del proceso de desarrollo de software.

La Máquina Virtual de Java está disponible en muchos sistemas operativos, lo que permite que los mismos archivos .class se ejecuten en diferentes plataformas (Figura 2). Además, algunas máquinas virtuales, como Java SE HotSpot, realizan pasos adicionales en tiempo de ejecución para mejorar el rendimiento de la aplicación (Oracle, 2023).



**Figura 2.** A través de Java VM, la misma aplicación es capaz de ejecutarse en múltiples plataformas.

Java es conocido por su facilidad de aprendizaje, la escritura de menos código en comparación con otros lenguajes, la promoción de buenas prácticas de codificación y la prevención de fugas de memoria gracias a la recolección automática de basura. También permite el desarrollo más rápido y la portabilidad, ya que las aplicaciones escritas en Java se pueden ejecutar en cualquier plataforma que admita Java (Oracle, 2023).

A continuación se muestra una comparación entre los lenguajes de programación Java, PHP y Python (ver Tabla 3), que son utilizados por diversas aplicaciones Web (Pardo et al., 2018).

**Tabla 3. Comparativa de los lenguajes de programación Java, PHP y Python**

<b>Característica</b>	<b>Java</b>	<b>PHP</b>	<b>Python</b>
Costo de servidor	Gratuito	Gratuito	Gratuito
Orientado a objetos	Sí	No completamente	Sí
Sistemas operativos	Linux o Windows	Linux o Windows	Linux o Windows
Servidor	Apache, Tomcat y Glassfish	Apache, compilador propio	Apache, compilador propio
Base de datos (principalmente)	Oracle, MySQL	MySQL	MySQL y PostgreSQL
Propósito	Generar dinámicamente páginas Web	Generar dinámicamente páginas Web	Enfatiza la productividad y la lectura fácil del código
Apoyo de aprendizaje	Mucha, pero descentralizada. No hay una entidad que de forma oficial centralice la ayuda	Mucha, pero descentralizada. No hay una entidad que de forma oficial centralice la ayuda	Mucha, pero descentralizada. No hay una entidad que de forma oficial centralice la ayuda
Ambiente de desarrollo	Eclipse, Netbeans y otras herramientas Open Source	Eclipse y otras herramientas Open Source	Eclipse, Netbeans y otras herramientas Open Source

### **2.3.3. JavaScript**

JavaScript fue creado por Brendan Eich en 1995 para Netscape Navigator, el navegador web dominante en ese momento. Originalmente se llamó Mocha y luego LiveScript antes de recibir su nombre actual. A pesar de su nombre, JavaScript no está relacionado con el lenguaje de programación Java, sino que la asociación se produjo debido a una colaboración entre Netscape y Sun Microsystems, el creador de Java (Boyd, 2023).

Hoy en día, JavaScript ha evolucionado más allá de ser solo un lenguaje de secuencias de comandos para la web. Es un lenguaje robusto y versátil que se utiliza en una variedad de formas, desde programación imperativa hasta programación orientada a objetos, enfoques funcionales y modelos orientados a eventos (Boyd, 2023).

JavaScript se utiliza en diversas aplicaciones, incluyendo (Boyd, 2023):

- Agregar contenido dinámico e interactivo a sitios web.
- Desarrollar aplicaciones web y móviles completas.
- Escribir código en el lado del servidor.
- Desarrollar videojuegos gracias a su versatilidad y capacidad para ejecutarse en diversos dispositivos.

La popularidad de JavaScript se debe a su ubicuidad en la web, su facilidad de aprendizaje, versatilidad y el amplio apoyo de la comunidad y las bibliotecas. Se utiliza en todos los principales navegadores web y es compatible con numerosas plataformas en la nube, lo que lo convierte en una elección poderosa para desarrolladores. Además, el salario promedio de un desarrollador de JavaScript en los Estados Unidos es atractivo (Boyd, 2023).

#### **2.3.4. PrimeFaces**

PrimeFaces es una suite de componentes de código abierto para JavaServer Faces (JSF) que ofrece una amplia variedad de extensiones y ventajas significativas. Incluye un conjunto completo de componentes, como HtmlEditor, Dialog, AutoComplete y Gráficos. Una característica destacada de PrimeFaces es su capacidad para utilizar AJAX integrado, aprovechando las API de AJAX estándar de JSF (PrimeFaces, 2023).

PrimeFaces es una elección ligera, ya que se distribuye en un solo archivo JAR sin requerir configuraciones adicionales ni dependencias externas (PrimeFaces, 2023).

En términos de seguridad, PrimeFaces se enorgullece de su prevención automática de ataques XSS (Cross-Site Scripting) y CSRF (Cross-Site Request Forgery), lo que brinda una capa adicional de protección para las aplicaciones. Además, PrimeFaces permite la sanitización personalizable del HTML mediante la utilización de la OWASP (Open Web Application Security Project) (PrimeFaces, 2023).

La suite PrimeFaces se destaca por su capacidad de respuesta y la accesibilidad, asegurando que las aplicaciones sean funcionales y utilizables en una amplia variedad de dispositivos y para usuarios con diversas necesidades (PrimeFaces, 2023).

PrimeFaces también ofrece un marco de temas con más de nueve temas integrados y soporte para una herramienta de diseño visual de temas. Además de los temas incorporados, PrimeFaces ofrece temas y diseños premium para aquellos que buscan opciones de personalización avanzada. La suite cuenta con una documentación extensa que ayuda a los desarrolladores a utilizar sus características de manera efectiva, y también se beneficia de una comunidad de usuarios grande, vibrante y activa (PrimeFaces, 2023).

En la Tabla 4 se comparan tres bibliotecas de uso común que se utilizan en la implementación JSF: Richfaces, IceFaces y Primefaces (Marchioni, 2021).



**Tabla 4. Comparación entre tres bibliotecas de uso común en la implementación JSF: Richfaces, IceFaces y Primefaces.**

<b>Característica</b>	<b>PrimeFaces</b>	<b>RichFaces</b>	<b>IceFaces</b>
Componentes disponibles	117 componentes (core + variantes), incluyendo HtmlEditors, gráficos, programador de fechas, exportador de datos Excel, etc.	39 componentes (11 core, 6 panels, 9 tables-grid, 1 tree, 4 toolbar, 8 menu)	70 componentes principales, componentes ACE adicionales
Facilidad de inicio	Página de inicio rápido disponible, fácil de comenzar con solo agregar el archivo JAR y la importación del espacio de nombres	Documentación enfocada en Maven, no tutorial de inicio rápido en la documentación oficial	Guía de inicio rápido disponible, se necesita un conjunto de bibliotecas principales y algunas dependencias
Documentación	Guía del usuario completa y práctica, con algunos recursos adicionales. Foro menos extenso en comparación con RichFaces e IceFaces.	Documentación en línea actualizada, con falta de tutoriales adicionales	Documentación extensa con tutoriales y videos. Registro necesario para acceder a ciertos recursos.
Problemas abiertos	Menos problemas abiertos, pero en un ciclo de vida y número de versiones diferente. Difícil de comparar de manera efectiva.	Más problemas abiertos, pero con un porcentaje menor de problemas críticos	Menos problemas abiertos, pero un porcentaje mayor de problemas críticos
Rendimiento	Mejor rendimiento en comparación con RichFaces e IceFaces, según las pruebas realizadas.	Menor rendimiento que PrimeFaces	Menor rendimiento que PrimeFaces

### 2.3.5. Spring Boot

Spring Boot es una herramienta que facilita la creación de aplicaciones basadas en Spring independientes y listas para producción, que pueden ejecutarse fácilmente. Spring Boot toma una visión opinada de la plataforma Spring y las bibliotecas de terceros, lo que permite comenzar con la configuración mínima. La mayoría de las aplicaciones de Spring Boot requieren una configuración mínima de Spring (Spring, 2023).

Las características clave de Spring Boot incluyen (Spring, 2023):

- Crear aplicaciones Spring independientes.
- Incrustar directamente servidores como Tomcat, Jetty o Undertow, eliminando la necesidad de desplegar archivos WAR.
- Proporcionar dependencias de “inicio” pensadas para simplificar la configuración de compilación.
- Configurar automáticamente Spring y bibliotecas de terceros siempre que sea posible.
- Ofrecer características listas para producción como métricas, comprobaciones de salud y configuración externalizada.
- No se requiere generación de código y no se necesita configuración XML.

Spring Boot acelera el desarrollo de aplicaciones al analizar su classpath y las configuraciones que ha establecido, haciendo suposiciones razonables sobre lo que falta y agregando automáticamente esos elementos. Con Spring Boot, los desarrolladores pueden centrarse más en las características del negocio y menos en la infraestructura (Spring, 2023).

Algunos ejemplos de lo que Spring Boot puede hacer automáticamente incluyen la configuración de contenedores Servlet como Tomcat o Jetty según lo que esté en el classpath y la adición de ciertos beans necesarios para el funcionamiento de Spring MVC o Thymeleaf. Estas configuraciones automáticas permiten que los desarrolladores se enfoquen en la lógica de la aplicación sin preocuparse por la configuración tediosa (Spring, 2023).

Spring Boot no genera código ni realiza ediciones en los archivos; en su lugar, dinámicamente conecta beans y configuraciones al contexto de la aplicación cuando se inicia, brindando un enfoque eficiente y sin complicaciones (Spring, 2023).

En la tabla que se muestra a continuación (ver Tabla 5), se realiza una comparación de los frameworks Java: Spring Boot, Micronaut y Quarkus (Jeleń & Dzieńkowski, 2021).

**Tabla 5. Comparación de los frameworks Java: Spring Boot, Micronaut y Quarkus.**

<b>Característica</b>	<b>Spring Boot</b>	<b>Micronaut</b>	<b>Quarkus</b>
Uso preferido	Microservicios	Microservicios	Aplicaciones nativas para Kubernetes
Lenguaje de	Java	Java, Scala, Kotlin	Java

programación			
Inicio rápido	Sí	Sí	Sí
Consumo de memoria	No especificado	Bajo consumo	Ligero en términos de tamaño y uso de memoria
Herramienta de construcción	Maven o Gradle	No especificado	Proporciona un proyecto configurado
Herramientas adicionales	Spring Data JPA, REST	No especificado	Integración con herramientas de contenedorización como Kubernetes

### 3. Metodología

La metodología de desarrollo del sistema seguirá un enfoque en cascada, es decir, lineal y secuencial, que incluye las siguientes etapas:

#### 3.1. Revisión bibliográfica

En esta etapa, se llevó a cabo una revisión exhaustiva de la literatura relacionada con el desarrollo de sistemas similares.

Se realizó una búsqueda de artículos en Google Académico para identificar los trabajos relacionados con el tema del sistema, obtener información sobre los requisitos funcionales y no funcionales de sistemas similares e identificar las mejores prácticas en el desarrollo de sistemas de gestión de producción científica.

Se buscaron artículos de los últimos 20 años que abordaran el tema de los sistemas de gestión de producción científica. Se leyó un total de 10 artículos, se leyó el resumen y los objetivos de cada uno para determinar si era relevante para el proyecto y se seleccionaron 6 que se relacionaban directamente con el proyecto.

#### 3.2. Recolección de Requisitos

La recolección de requisitos se llevó a cabo mediante entrevistas con el orientador y co-orientador del proyecto, quienes son profesores del IFSul y conocen las necesidades de la institución.

Se identificaron las necesidades de los usuarios y se analizó la información recopilada para definir los requisitos funcionales y no funcionales del sistema (ver Apéndice I y II).

### **3.3. Implementación**

La fase de diseño de la interfaz de usuario del sistema se realizará utilizando un enfoque de diseño centrado en el usuario teniendo en cuenta los principios de Jakob Nielsen (1993, citado en Sánchez, 2011).

Se buscará la creación de una interfaz de usuario que sea limpia, fácil de usar y altamente intuitiva para los usuarios, utilizando el lenguaje de los mismos, con íconos, palabras, frases y conceptos familiares a ellos, en lugar de los términos orientados al sistema, reduciendo al mínimo la información que no sea aplicable o que raramente se necesite así como la cantidad de pasos necesarios para acceder a cualquier parte del sistema. Además, se implementará una interfaz que permita mantener a los usuarios informados del estado del sistema e identificar fácilmente su ubicación en el sistema en todo momento.

Se realizará el modelado de la base de datos que respaldará el sistema utilizando el estándar ER (Entity-Relationship) y luego convertirlo al modelo lógico (ver Apéndice III). Se utilizará una base de datos relacional MySQL para almacenar los datos del sistema debido a su robustez y capacidad para trabajar con bases de datos no demasiado grandes (LovTechnology, 2023), además de ser open source y de código abierto, lo que significa que es gratuita y que su código está disponible para que cualquiera lo revise y modifique (MySQL, 2023).

El desarrollo del sistema se llevará a cabo utilizando diversos lenguajes y tecnologías. El núcleo del sistema se construirá utilizando Java porque es un lenguaje de programación general que es adecuado para desarrollar aplicaciones web. Es un lenguaje de programación robusto y maduro que tiene una gran comunidad de desarrolladores (Oracle, 2023). Además, se utilizará JavaScript porque es un lenguaje de programación ligero y fácil de aprender que se utiliza para agregar interactividad a las páginas web (Pluralsight, 2023).

Se aprovechará las capacidades de PrimeFaces para los componentes de la interfaz porque, además de ser específico para Java para crear interfaces de usuario modernas y receptivas, es de código abierto y gratuito (PrimeFaces, 2023).

Además, se empleará Spring Boot porque es un marco de trabajo de desarrollo web que facilita el desarrollo de aplicaciones web Java y garantiza una sólida arquitectura del sistema. Es un marco de trabajo ligero y fácil de aprender (Spring, 2023).

### **3.4. Pruebas**

Las pruebas del sistema se llevarán a cabo mediante varios tipos de pruebas en base al modelo en V. Se realizarán pruebas unitarias para cada método del sistema, garantizando su correcto funcionamiento. Las pruebas de integración se centrarán en los módulos y/o servicios que interactúen entre sí para verificar que los mismos funcionen correctamente juntos. Se llevarán a cabo pruebas de sistema simulando ser un usuario real para evaluar la experiencia del usuario y se realizarán pruebas de aceptación con la participación de los stakeholders después de la implementación del sistema para evaluar las necesidades de los usuarios.

### 3.5. Implantación

La implementación del sistema se llevará a cabo en múltiples entornos. El sistema estará disponible tanto en la red pública como en la red privada de IFSul. La visualización de la herramienta desarrollada o su código fuente estará restringida a la red privada de IFSul para garantizar la seguridad y confidencialidad. Sin embargo, las demás funcionalidades del sistema estarán disponibles para su acceso desde cualquier red (ver Tabla 6).

**Tabla 6. Funcionalidades del sistema propuesto y su accesibilidad en diferentes redes**

<b>Funcionalidad del sistema</b>	<b>Descripción</b>	<b>Accesibilidad en redes</b>
Catálogo de producción científica	Ingreso, edición y eliminación de producción científica	Accesible desde cualquier red con inicio de sesión
	Búsqueda por título, área, período, orientador, autor, curso y palabras clave.	Accesible desde cualquier red sin necesidad de inicio de sesión.
	Acceso al tema, título, área, resumen, PDF del trabajo, autor, orientador, co-orientador y trabajos previos.	Accesible desde cualquier red sin necesidad de inicio de sesión.
	Visualización de la herramienta desarrollada o su código fuente.	Accesible desde la red privada de IFSul con inicio de sesión
Base de datos de temas de TCC y demandas de orientadores	CRUD de temas de TCC	Accesible desde cualquier red con inicio de sesión
	Selección de temas de TCC.	Accesible desde cualquier red con inicio de sesión
	Solicitud de orientador y generación de término de compromiso.	Accesible desde cualquier red con inicio de sesión
Acceso a repositorios de otras universidades	Enlaces a repositorios de otras universidades.	Accesible desde cualquier red sin necesidad de inicio de sesión
Administración de usuarios	CRUD de usuarios del sistema	Accesible desde cualquier red con inicio de sesión

### 3.6. Documentación

En esta etapa se proporcionará una documentación completa que incluye:

1. Documento de Requisitos: Este documento contendrá una lista detallada de los requisitos funcionales y no funcionales del sistema. Estos requisitos servirán como guía para el diseño, desarrollo y evaluación del sistema.
2. Diagramas de Caso de Uso: Se incluirán diagramas que ilustran cómo los diferentes actores interactúan con el sistema y las funciones a las que pueden acceder. Estos diagramas proporcionarán una visión general de la interacción usuario-sistema.
3. Modelo Lógico: Este diagrama mostrará la estructura de la base de datos utilizada en el sistema, lo que ayudará a comprender la organización de los datos y las relaciones entre las entidades.
4. Informes de Pruebas: Los informes detallarán los resultados de las pruebas realizadas en el sistema, incluyendo pruebas unitarias, de integración, de sistema y de aceptación. Estos informes servirán como evidencia de que el sistema se probó exhaustivamente y cumple con los requisitos especificados.

Esta documentación proporcionará información detallada sobre el sistema y su funcionamiento, lo que facilitará su implementación y futuras actualizaciones. Además, servirá como referencia para cualquier persona que necesite comprender o trabajar con el sistema en el futuro.

## 4. Cronograma

El siguiente cronograma detalla las fases y actividades planificadas para desarrollo del Sistema de Gestión de Producción Científica del IFSul Campus Santana do Livramento. Este cronograma permitirá coordinar eficientemente los esfuerzos en la etapa de desarrollo, asegurando así la implementación exitosa del sistema y el cumplimiento de los objetivos establecidos en este proyecto.

[illegible]

## Referencias

- Almeida, I. R., Oliveira, B. M. J. F., & ROSA, M. N. B. (2019). Repositórios digitais como espaços de memória e disseminação de informação. *Informação em Pauta*, 4(n. especial), 117-131.
- Amorim, F. M. D. (2015). Repositório digital para o acervo de trabalho de conclusão de curso da Biblioteca David Sá.
- Boyd, W. (2023). What is the Javascript programming language?. Pluralsight. Recuperado el 31 de octubre de 2023 de <https://www.pluralsight.com/resources/blog/cloud/what-is-the-javascript-programming-language#:~:text=History%20of%20JavaScript,users%20interacted%20with%20the%20site>
- Da Silva, L. F. S. (2023). A importância de um acervo com projetos de pesquisa de TCC no Instituto Federal da Paraíba (Bachelor's thesis).
- Faculdade Santíssima Trindade. (s.f.). Manual de orientações para Trabalho de Conclusão e Curso TCC. Recuperado el 17 de octubre de 2023 de: <https://edufast.com.br/wp-content/uploads/2022/01/MANUAL-DE-TCC.pdf>.
- Freitas, T. C. S. (2012). A percepção dos discentes sobre as dificuldades na produção do trabalho acadêmico. En IX Anped Sul, Seminário de pesquisa em educação da região sul. Recuperado el 17 de octubre de 2023 de <https://docplayer.com.br/9064818-A-percepcao-dos-discentes-sobre-as-dificuldades-na-producao-do-trabalho-academico.html>
- IFSul Santana Do Livramento. (2023). Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas - Projeto Pedagógico do Curso. Recuperado 17 de octubre de 2023 de: <https://intranet.ifsul.edu.br/catalogo/download/projeto/770>
- Java. (s.f.). What is Java technology and why do I need it?. Recuperado el 31 de octubre de 2023 de [https://www.java.com/en/download/help/whatis\\_java.html](https://www.java.com/en/download/help/whatis_java.html)
- Jeleń, M., & Dzieńkowski, M. (2021). The comparative analysis of Java frameworks: Spring Boot, Micronaut and Quarkus. *Journal of Computer Sciences Institute*, 21, 287-294.
- Lenoir, Y. (2006). Pesquisar e formar: repensar o lugar e a função da prática de ensino. *Educação & Sociedade*, 27(97), 1299-1325.
- LovTechnology. (s.f.). MySQL vs PostgreSQL vs SQLite: Comparación detallada entre estos sistemas de gestión de base de datos relacionales. Recuperado el 19 de noviembre de 2023 de <https://lovtechnology.com/mysql-vs-postgresql-vs-sqlite-comparacion-detallada/>
- Marchioni, F. (2021). PrimeFaces vs RichFaces vs IceFaces. Mastertheboss. Recuperado el 20 de noviembre de 2023 de <https://www.mastertheboss.com/web/richfaces/primefaces-vs-richfaces-vs-icefaces/>
- Marques, N. C. (2019). Desafios vivenciados pelos licenciados em biologia da UFRB na construção do projeto de pesquisa para o trabalho de conclusão de curso. Cruz das Almas, BA.

Murakami, T. R. M., & Fausto, S. (2013). Panorama atual dos Repositórios Institucionais das Instituições de Ensino Superior no Brasil. *Incid*, 4(2), 185-201.

MySQL. (s.f.). History of MySQL. Recuperado el 31 de octubre de 2023 de <https://dev.mysql.com/doc/refman/8.0/en/history.html>

MySQL. (s.f.). The Main Features of MySQL. Recuperado el 31 de octubre de 2023 de <https://dev.mysql.com/doc/refman/8.0/en/features.html>

MySQL. (s.f.). What is MySQL?. Recuperado el 31 de octubre de 2023 de <https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html>

Oracle. (s.f.). About the Java Technology. Recuperado el 31 de octubre de 2023 de <https://docs.oracle.com/javase/tutorial/getStarted/intro/definition.html>

Oracle. (s.f.). How Will Java Technology Change My Life?. Recuperado el 31 de octubre de 2023 de <https://docs.oracle.com/javase/tutorial/getStarted/intro/changemylife.html>

Palomino, N. L. S., Vásquez, A. C., & Jaime, F. G. (2010). Propuesta de desarrollo de un repositorio digital de documentos de investigación para la FISI utilizando software libre. *Revista de investigación de Sistemas e Informática*, 7(2), 69-75.

Pardo, M. R. V., Tapia, J. A. H., Moreno, A. S. G., & Sánchez, L. F. V. (2018). Comparación de tendencias tecnológicas en aplicaciones web. *3c Tecnología: glosas de innovación aplicadas a la pyme*, 7(3), 28-49.

Pavão, C. M. G., Araujo Neto, A. C. D., Caregnato, L. F., Costa, J. S. B. D., Horowitz, Z., Oliveira, Z. C. P. D., & Saatkamp, C. M. (2008). Repositório digital: acesso livre à informação na Universidade Federal do Rio Grande do Sul. In *Conferência Ibero-Americana de Publicações Eletrônicas no Contexto da Comunicação Científica (2.: 2008: Rio de Janeiro)*. Anais. Rio de Janeiro:[sn], 2008..

Pereira, A., & Silva, M. de L. (s.f.). O Trabalho de conclusão de curso: constructo epistemológico no currículo formação, valor e importância. Recuperado el 17 de octubre de 2023 de <https://docplayer.com.br/10778788-O-trabalho-de-conclusao-de-curso-constructo-epistemologico-no-curriculo-formacao-valor-e-importancia.html>

PrimeFaces. (s.f.). PrimeFaces Documentation V11.0.0. Recuperado el 31 de octubre de 2023 de [https://primefaces.github.io/primefaces/11\\_0\\_0/#/](https://primefaces.github.io/primefaces/11_0_0/#/)

Ramírez, M. R., Soto, M. D. C. S., Moreno, H. B. R., Rojas, E. M., Millán, N. D. C. O., & Cisneros, R. F. R. (2019). Metodología SCRUM y desarrollo de Repositorio Digital. *Revista Ibérica De Sistemas e Tecnologias De Informação*, (E17), 1062-1072.

Rodrigues, E., Almeida, M., Miranda, Â., Guimarães, A. X., & Castro, D. (2004). RepositóriUM: criação e desenvolvimento do Repositório Institucional da Universidade do Minho.

Sánchez, W. (2011). La usabilidad en Ingeniería de Software: definición y características. *Ing-novación. Revista de Ingeniería e Innovación de la Facultad de Ingeniería, Universidad Don Bosco*, 1(2), 7-21. ISSN 2221-1136.



Silva, L. N. D. (2022). Uma proposta de desenvolvimento de um acervo digital para o app itinerante fácil TCC do Campus Teresina Zona Sul.

Spring. (s.f.). Building an Application with Spring Boot. Recuperado el 31 de octubre de 2023 de <https://spring.io/guides/gs/spring-boot/>

Spring. (s.f.). Spring Boot. Recuperado de 31 de octubre de 2023 de <https://spring.io/projects/spring-boot>

Texier, J., De Guisti, M., & Gordillo, S. (2014). El desarrollo de software dirigido por modelos en los repositorios institucionales. *Dyna*, 81(184), 186-192.

Viana, C. L. M., & Arellano, M. A. M. (2006). Repositórios Institucionais baseados em Dspace e Eprints e sua visibilidade nas instituições acadêmico-científicas. En *Seminário Nacional de Bibliotecas Universitárias. Anais* (pp. 1-15). Salvador: UFBA.

## **Apéndice I: Documento de requisitos**

### **Documento de Requisitos del Sistema**

#### **Sistema de Gestión de Producción Científica del IFSul Campus Santana do Livramento**

##### **Introducción**

Este documento describe los requisitos del sistema para el "Sistema de Gestión de Producción Científica del IFSul Campus Santana do Livramento". El sistema tiene como objetivo principal proporcionar una plataforma para la gestión, búsqueda y visualización de la producción científica, así como la administración de temas de Trabajos de Conclusión de Curso (TCC) en el contexto del Instituto Federal de Educação, Ciência e Tecnologia Sul-riograndense (IFSul) Campus Santana do Livramento. Además, permite el acceso a repositorios de otras universidades.

##### **Requisitos Funcionales**

###### **1. Catálogo de producción científica**

- 1.1. El sistema debe permitir a los usuarios buscar producciones científicas por título, área, período, orientador, autor, curso y palabras clave.
- 1.2. Los usuarios deben poder ver los resultados de búsqueda, incluyendo título, autor, resumen y palabras clave.
- 1.3. Al hacer clic en un resultado, se debe mostrar información detallada, como título, autor, área, resumen, PDF del trabajo, palabras clave, autor, orientador, co-orientador (si corresponde) y trabajos previos.
- 1.4. Los administradores deben poder mantener nuevas producciones científicas ya aprobadas

###### **2. Base de datos de temas de TCC**

- 2.1. Los administradores deben mantener nuevos temas de TCC.
- 2.2. Los estudiantes deben poder seleccionar temas de TCC o proponer un nuevo tema.
- 2.3. Los estudiantes deben poder solicitar orientadores para sus temas de TCC.
- 2.4. Los orientadores deben poder responder a la solicitud de orientación.

2.5. El sistema debe generar automáticamente un "término de compromiso" en PDF para formalizar la selección del tema del TCC y el orientador.

### 3. Acceso a repositorios de otras universidades

3.1. El sistema debe incluir enlaces a repositorios de otras universidades con sistemas similares.

## **Requisitos No Funcionales**

### 1. Rendimiento

1.1. El sistema debe ser capaz de manejar una carga de trabajo significativa sin degradación del rendimiento, permitiendo la consulta y recuperación de datos de manera eficiente.

### 2. Seguridad

2.1. El acceso a la base de datos y a la información sensible del sistema debe estar protegido por autenticación y autorización adecuadas.

2.2. El sistema debe garantizar la confidencialidad de los datos de los estudiantes, autores y orientadores.

2.3. Se debe realizar copias de seguridad regulares de la base de datos para garantizar la integridad y disponibilidad de los datos.

### 3. Usabilidad

3.1. La interfaz de usuario debe ser intuitiva y fácil de usar, minimizando la curva de aprendizaje para los usuarios finales.

3.2. El sistema debe ser accesible desde diferentes dispositivos y navegadores web, garantizando una experiencia de usuario consistente.

### 4. Disponibilidad

4.1. El sistema debe estar disponible las 24 horas del día, los 7 días de la semana, con tiempos de inactividad planificados para el mantenimiento.

4.2. Se deben implementar medidas de recuperación ante desastres para garantizar la continuidad del servicio en caso de fallos.

### 5. Escalabilidad

5.1. El sistema debe ser escalable para manejar un aumento en el número de usuarios y datos sin problemas.

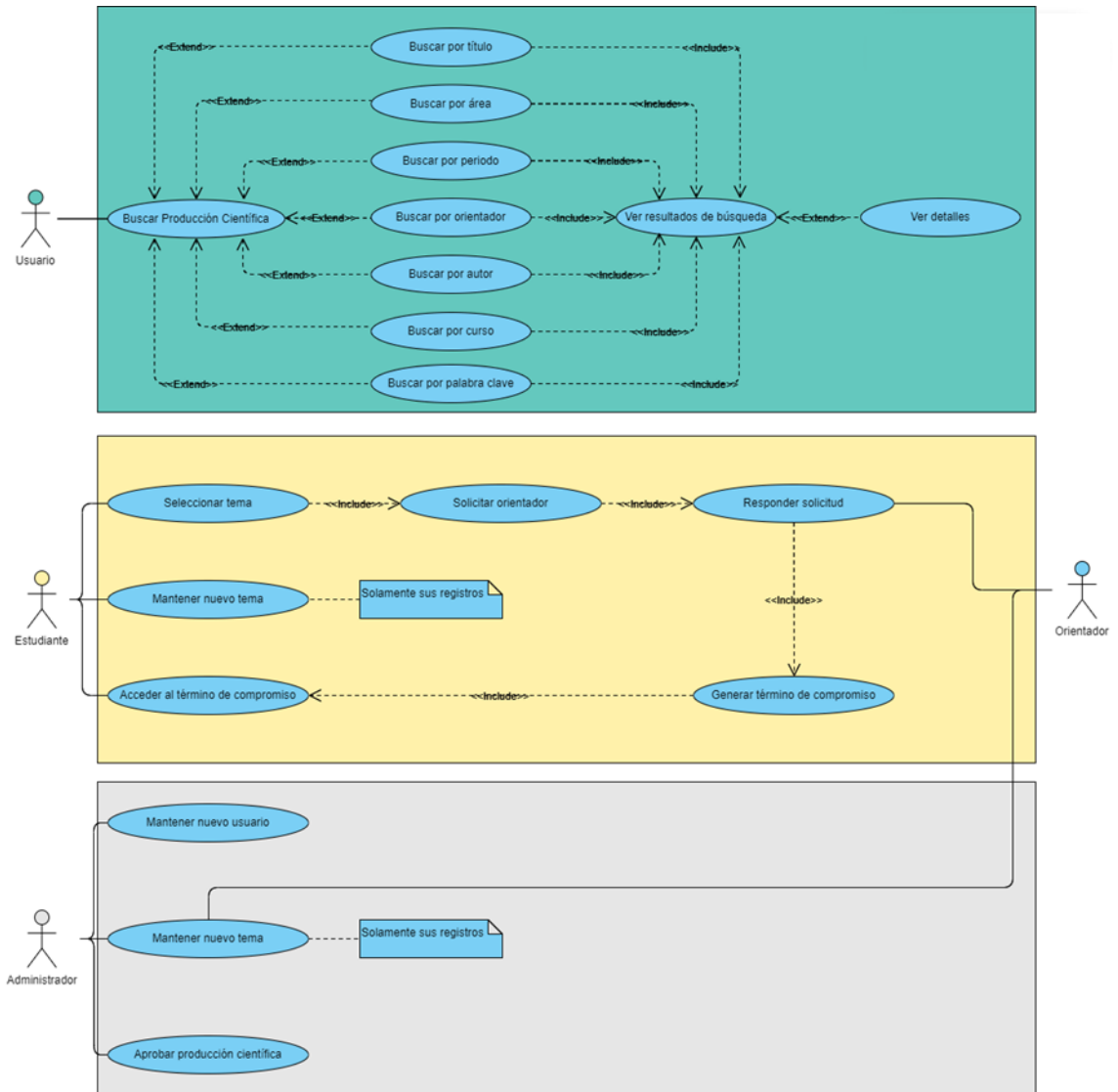
## 6. Idiomas

6.1. El sistema debe admitir múltiples idiomas, incluyendo portugués y español, para satisfacer las necesidades de usuarios de diferentes regiones.

## 7. Cumplimiento Legal

7.1. El sistema debe cumplir con todas las regulaciones y leyes locales relacionadas con la gestión de datos y privacidad.

## Apéndice II: Diagramas de casos de uso



## Apéndice III: Modelo lógico de la base de datos

