



Tecnológico de Monterrey

Planeación de sistemas de software

Grupo 106

Software Requirements Specifications

Profesores:

Adan Octavio Ruiz Martinez

Jorge Alvarez Bujanos

Leonardo S. Gamez Peña

José Francisco Mendoza Bazán

Equipo 6:

Eugenia Uresti Arriaga A01284839

Mónica Parra Franco A01251941

Gerardo Manzur Morales A01177622

Jorge Nuñez Gurrola A00833455

10 de marzo de 2024

1	Introducción.....	3
1.1	Propósito.....	3
1.2	Alcance.....	3
1.3	Definiciones, Acrónimos y Abreviaturas.....	3
1.4	Referencias.....	3
1.5	Visión General del Documento.....	4
2	Descripción General.....	4
2.1	Perspectiva del producto.....	4
2.1.1	Interfaces del Sistema.....	4
2.1.2	Interfaces del Usuario.....	4
2.1.3	Interfaces del Hardware.....	5
2.1.4	Interfaces del Software.....	5
2.1.5	Interfaces de Comunicación.....	5
2.1.6	Restricciones de memoria.....	5
2.1.7	Operaciones.....	6
2.1.8	Requisitos de adaptación del sitio.....	6
2.2	Funciones del producto.....	6
2.3	Características del usuario.....	6
2.4	Limitaciones.....	7
2.5	Suposiciones y dependencias.....	7
2.6	Asignación de requisitos.....	7
3	Requisitos específicos.....	8
3.1	Requerimientos funcionales.....	8
3.2	Requerimientos no funcionales.....	11
3.3	Restricciones de diseño.....	13
3.3.1	Cumplimiento de estándares.....	13
3.4	Atributos del sistema de software.....	13
3.4.1	Confiabilidad.....	13
3.4.2	Disponibilidad.....	13
3.4.3	Seguridad.....	14
3.4.4	Mantenibilidad.....	14
3.4.5	Portabilidad.....	14

1 Introducción

1.1 Propósito

El propósito de este Documento de Especificaciones de Requerimientos de Software (SRS) es definir los requisitos para el Oracle Java Bot, un chatbot diseñado para mejorar un 20% la productividad y visibilidad de las actividades de los desarrolladores y managers en Oracle. Este documento está destinado a ser utilizado por los miembros del equipo de desarrollo, el socio formador del proyecto y otras partes interesadas para asegurar una comprensión clara y un acuerdo sobre los objetivos del proyecto acordados antes del inicio del desarrollo.

1.2 Alcance

El producto de software a ser producido es el Oracle Java Bot, un servicio de chatbot que automatiza la gestión de tareas para los desarrolladores y proporciona visibilidad al mánager sobre las tareas del equipo. El bot operará dentro de la plataforma de mensajería Telegram y se integrará con la infraestructura de Oracle Cloud y Oracle Autonomous Database. Este bot permitirá a los desarrolladores agregar, revisar, modificar y eliminar sus tareas, mientras que el mánager tendrá las mismas acciones, pero con la capacidad de visualizar las tareas de todo el equipo bajo su responsabilidad.

1.3 Definiciones, Acrónimos y Abreviaturas

- DevOps: Prácticas y herramientas destinadas a aumentar la capacidad de una organización para entregar aplicaciones y servicios a alta velocidad.
- SRS: Software Requirements Specification.
- API: Application Programming Interface.
- OCI: Oracle Cloud Infrastructure.
- OAD: Oracle Autonomous Database.

1.4 Referencias

- IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications.
- Documentación API de Telegram. [Telegram](#)
- Documentación de Oracle Cloud Infrastructure (OCI) [OCI](#)
- Documentación de Docker [Docker](#)
- Documentación de Java [Java](#)
- Documentación de Spring Boot [Spring Boot](#)
- Documentación de Java SE Development Kit (JDK) 8 [JDK 8](#)
- Documentación de Apache Maven [Maven](#)

1.5 Visión general del documento

El resto del documento detalla los requisitos específicos del Oracle Java Bot, organizados por funcionalidades, restricciones de diseño, atributos del sistema de software y otros requerimientos. La Sección 2 proporciona una descripción general del producto, mientras que la Sección 3 detalla los requisitos específicos.

2 Descripción general

2.1 Perspectiva del producto

El Oracle Java Bot es un producto independiente diseñado para integrarse con la infraestructura existente de Oracle Cloud y la plataforma de Telegram. Funciona como un componente clave en la estrategia de DevOps del equipo de desarrollo, facilitando la automatización y gestión de tareas.

2.1.1 Interfaces del sistema

- Oracle Cloud Infrastructure (OCI)

El Oracle Java Bot se integrará con Oracle Cloud infrastructure para facilitar el despliegue y las operaciones, aprovechando las funcionalidades de la infraestructura en la nube Oracle.

- Oracle Autonomous Database

El bot gestiona información relacionada con los usuarios y sus tareas almacenando los datos en Oracle Autonomous Database, garantizando la integridad y la persistencia de los datos.

- Telegram

El Oracle Java Bot interactuara y permitirá la visualización y la gestión de las tareas a través de Telegram.

2.1.2 Interfaces del usuario

La interfaz del usuario está basada en texto para interactuar con el bot, incluyendo comandos y botones específicos para agregar, listar, modificar y eliminar tareas. Se optimizará la usabilidad y accesibilidad, ofreciendo mensajes de error claros y concisos, junto con atajos/sugerencias para la siguiente acción a realizar. Mientras que los desarrolladores podrán visualizar y gestionar solamente sus tareas, los managers podrán gestionar y visualizar las tareas de todo su equipo.

2.1.3 Interfaces del hardware

Para el proyecto Oracle Java ChatBot, aunque se aprovecha principalmente la infraestructura en la nube a través de OCI, es esencial considerar la interacción con el hardware del usuario final, incluidos dispositivos como computadoras y teléfonos móviles, que son cruciales para acceder y operar el bot. Este acceso indirecto subraya la relevancia de los servidores de OCI, cuya capacidad y rendimiento (procesamiento, memoria, almacenamiento) son fundamentales para asegurar una experiencia de usuario fluida y efectiva en sus dispositivos personales.

2.1.4 Interfaces del software

- Java y Spring Boot

Estas herramientas se utilizan para el desarrollo del bot, ya que garantizan un entorno de ejecución eficiente y compatible.

Version: Java JDK 11 y Spring Boot 2.5.

- Telegram Bot API.

Esta herramienta facilita la comunicación entre el bot y los usuarios a través de Telegram.

Version: Ultima version compatible.

- Oracle autonomous Database.

Se utiliza este componente tanto para almacenar como para gestionar los datos del Oracle Java Bot.

Version: Ultima version compatible.

2.1.5 Interfaces de comunicación

- HTTPS

Para asegurar la confidencialidad e integridad de los datos durante la comunicación, se utiliza HTTPS para garantizar una transmisión segura de datos entre el bot, la plataforma de Telegram y la base de datos en Oracle Autonomous Database.

2.1.6 Restricciones de memoria

Ya que la interfaz del ChatBot está alojada en la nube, no habría restricciones de memoria para los usuarios finales; aunque, es importante que el OCI asignado tenga suficiente memoria para soportar la lógica del bot y el manejo de múltiples instancias de usuario al mismo tiempo, esto dependerá del número de usuarios y la complejidad de las operaciones que deberá realizar el bot.

2.1.7 Operaciones

- Modos de operación: Interacción en tiempo real con usuarios a través de Telegram para gestión y visualización de tareas.
- Backup y Recuperación: Procesos automatizados para asegurar la integridad de datos.

2.1.8 Requisitos de adaptación del sitio

El bot debería ser capaz de adaptarse a cualquier tipo de sitio donde se implemente, esto mientras el entorno donde se aplique cumpla con los requisitos de infraestructura de OCI; también existe la posibilidad de necesitar una conexión a Internet para la comunicación entre las APIs de Telegram y la Oracle Autonomous Database (OAD), para ello tendrían que configurarse el acceso de red para que esta sea segura.

2.2 Funciones del producto

Dentro del sistema todos los desarrolladores tendrán la oportunidad de tener una interacción con el bot a través de comandos específicos en Telegram, esto con el propósito de mejorar la productividad y visibilidad de las actividades de cada uno. El bot permitirá a los desarrolladores listar, agregar, modificar y eliminar sus tareas, como también se les permitirá marcar tareas como completadas o volver a reactivar tareas previamente completadas. Como manager de equipo, también se les permitirá listar, agregar, modificar y eliminar tareas de todo el equipo bajo su responsabilidad, de igual forma tendrán la oportunidad de marcar tareas como completadas o de volver a reactivar tareas ya marcadas como completadas.

2.3 Características del usuario

Este proyecto involucra a dos grupos principales, donde ambos requieren ser parte de proyectos de Oracle, donde tengan la capacidad de adaptarse rápidamente a nuevas herramientas y a estar buscando constantemente oportunidades para aprender y crecer profesionalmente.

- Managers

Como manager requieren tener la capacidad de interactuar eficientemente con el equipo para poder asignar tareas y dar un buen seguimiento al proyecto, como también se necesitan de habilidades efectivas de comunicación para poder liderar y orientar al equipo sobre el proyecto. Se necesita de la capacidad de analizar las fortalezas y debilidades individuales de cada miembro del equipo para así poder asignar tareas de manera estratégica y optimizar aún más el rendimiento de cada uno.

- Desarrolladores

Como desarrollador requieren de la capacidad de priorizar sus tareas para cumplir con los plazos de entrega y necesitan de la habilidad de trabajar y comunicarse efectivamente con otros miembros del equipo.

2.4 Limitaciones

- La comunicación entre el equipo y el servicio debe de ser a través de mensajes de Telegram para garantizar la seguridad.
- La arquitectura del software debe ser Cloud Native, utilizando Oracle Cloud Infrastructure, Oracle Autonomous Database, Kubernetes, Docker y Spring Boot framework.
- El bot debe operar dentro de los límites de seguridad establecidos para la comunicación a través de Telegram.
- El sistema debe de tener un proceso de autenticación por contraseña para los usuarios.
- Toda información transmitida y guardada en la base de datos debe ser encriptada para proteger la información de los usuarios.

2.5 Suposiciones y dependencias

- Se asume que los usuarios contarán con una conexión a internet para utilizar ChatBot.
- Se asume que los usuarios tendrán acceso a los recursos necesarios para utilizar el ChatBot.
- Se asume que el equipo de desarrollo tendrá acceso a las herramientas necesarias para la implementación exitosa del ChatBot.
- El sistema depende de la API de Telegram para la interacción entre el ChatBot y los usuarios.
- La infraestructura del sistema depende de los servicios de Oracle Cloud Infrastructure.

2.6 Asignación de requisitos

- Particionamiento funcional
 - El chatbot tendrá requisitos para funciones específicas para la gestión y almacenamiento de tareas. Entre estos varios comandos relacionados con las tareas ya sea la extracción de información relevante y la capacidad de almacenarlas

- Tendrá también módulos para identificar cuando una tarea está pendiente, es completada o fue eliminada.
- Asignación arquitectónica
 - Tendrá la arquitectura de una base de datos donde se almacenarán las tareas creadas.
- Asignación de interfaces
 - El chatbot tendrá varios botones en los cuales tendrá la oportunidad de elegir si se quisiera agregar una nueva tarea, Ver las tareas pendientes, Marcar tareas como completadas y borrar tareas
- Asignación de rendimiento
 - Tendrá requisitos de rendimiento para así poder llevar a cabo una gran número creciente de tareas y poder proporcionar respuestas rápidas
- Asignación de datos
 - Tendrá requisitos para la gestión en la estructura de tareas ya sea para la información, prioridad, porcentaje completado, etc
 - También definirán la persistencia y recuperación de los datos de la tarea

3 Requisitos específicos

3.1 Requerimientos funcionales

Número	Nombre	Descripción	Prioridad
RF-01	Inicio de sesión	Incluye dos campos para ingresar usuario y contraseña.	Alta
RF-02	Mensajes informativos	Se muestran mensajes claros sobre el estado de inicio de sesión, como "Contraseña incorrecta" o "Usuario Incorrecto"	Media
RF-03	Campo de texto	Incluye un campo de texto en la	Alta

		pantalla principal donde los usuarios podrán ingresar lo que deseen hacer.	
RF-04	Botón para enviar	Se incluye a lado de el campo de texto un botón para enviar el comando que se haya escrito.	Alta
RF-05	Secciones de tareas	Al mostrar el listado de tareas, se muestra en la parte superior las tareas sin completar y en la parte inferior se muestran las tareas ya completadas.	Media
RF-06	Agregar tarea	Botón que agrega una nueva tarea.	Alta
RF-07	Eliminar tarea	Botón que te permite eliminar una tarea del listado	Alta
RF-08	Completar tarea	Botón que te permite marcar como completada una tarea	Alta
RF-09	Reactivación de tarea	Botón que permite reactivar una tarea ya completada, es decir regresar tarea a no completada	Alta

RF-10	Modificar tarea	Botón que te permite modificar alguno de los campos de una tarea.	Alta
RF-11	Sugerencias	Después de cada comando se muestran mensajes con atajos/sugerencias para el siguiente comando.	Media
RF-12	Redirección a pantalla principal	Botón que te permite regresar a la pantalla principal	Alta
RF-13	Entradas incorrectas	Si el usuario llega a insertar comandos incorrectos, el bot responde con ejemplos de comandos disponibles.	Media
RF-14	Filtrar tareas	Opciones de filtro que ayudan a los usuarios a ver tareas específicas.	Bajo
RF-15	Listar tareas de managers	Botón que enlista en dos secciones todas las tareas de todo el equipo bajo su responsabilidad.	Alta
RF-16	Nueva tarea de	Al agregar una	Alta

	managers	nueva tarea el bot pide diferentes campos a llenar, como título, categoría, descripción, prioridad, estatus, fecha fin y responsable.	
RF-17	Listar tareas de desarrollador	Botón que enlista en dos secciones todas las tareas asignadas al usuario.	Alta
RF-18	Nueva tarea para desarrollador	Al agregar una nueva tarea el bot pide diferentes campos a llenar, como título, categoría, descripción, prioridad, estatus y fecha de entrega.	Alta

3.2 Requerimientos no funcionales

Número	Nombre	Descripción	Prioridad
RNF-01	Usabilidad	El sistema debe de ser intuitivo y fácil de usar	Alta
RNF-02	Conexión con Oracle Autonomous Data Base	Debe de existir una conexión con Oracle Autonomous	Alta

		Database donde se guardará la información de los usuarios	
RNF-03	Seguridad	Toda la información guardada en la base de datos debe ser encriptada	Alta
RNF-04	Rendimiento	El sistema debe de ser capaz de manejar simultáneamente al menos a 60 usuarios	Alta
RNF-05	Disponibilidad	El sistema debe de estar disponible las 24 horas del día	Alta
RNF-06	Comunicación	Toda la comunicación entre los usuarios y el servicio deben de ser a través de Telegram	Alta
RNF-07	Arquitectura Cloud Native	El sistema debe de ser una aplicación nativa de la nube, utilizando Oracle Cloud Infrastructure, Oracle Autonomous Database y Java.	Alta

3.3 Restricciones de diseño

- Uso exclusivo de Telegram debido a restricciones de seguridad para la comunicación.
- El sistema debe ser desarrollado en Java JDK.
- El sistema debe de cumplir con las regulaciones y normativas en cuanto a privacidad y seguridad de datos.
- El ChatBot debe de integrarse con Oracle Autonomous Database para el almacenamiento de datos.

3.3.1 Cumplimiento de estándares

- El producto final debe cumplir con el estándar IEEE Std 830-1998, el cual considera recomendaciones prácticas para la especificación de requerimientos.
- El estándar de codificación del producto final en Java debe asegurar la legibilidad y mantenibilidad del código aplicado.
- Deben seguirse las directrices de la API de telegram para la implementación e integración de un bot.
- Deben aplicarse los estándares de Oracle para la integración de OCI y OAD.

3.4 Atributos del sistema de software

3.4.1 Confiabilidad

El sistema deberá tener una tasa de error inferior al 1% para asegurar una ejecución confiable de sus funciones, para esto se realizarán diferentes pruebas de calidad, para identificar y corregir posibles errores.

3.4.2 Disponibilidad

El sistema debe de garantizar una disponibilidad del 99.9%, lo que significa que el sistema estará activo mínimo durante el 99.9% del tiempo total. Para lograr esto se utilizarán estrategias como el mecanismo de checkpoint y recuperación, para garantizar que los usuarios puedan acceder al sistema de manera constante.

3.4.3 Seguridad

Para garantizar la confidencialidad en el sistema se encriptará todos los datos almacenados en la base de datos y se implementará el cifrado de extremo a extremo para las comunicaciones con el ChatBot.

3.4.4 Mantenibilidad

El código del sistema seguirá los principios de diseño modular, para facilitar la comprensión y el mantenimiento del sistema sin afectar el funcionamiento del sistema.

3.4.5 Portabilidad

El sistema será desarrollado en Java, por lo que se garantiza así su portabilidad en diferentes sistemas operativos y entornos de ejecución de Java.