

UNIVERSIDAD DE TARAPACÁ



UNIVERSIDAD DE TARAPACÁ
Universidad del Estado

FACULTAD DE INGENIERÍA

Departamento de Ingeniería en Computación e Informática



Avance 2

**“Sistema de Gestión y Control de
Presupuesto para la Facultad de
Odontología de la Universidad de
Chile”**

Autor(es): Diego Baltazar Fernández

Dylan Choque Nina

Asignatura : Proyecto IV

Profesor: Diego Aracena Pizarro

Cliente: Roberto Gamboa Aguilar

Arica, 4 de Junio de 2024

ÍNDICE DE CONTENIDO.

| | |
|--|-----------|
| 1. INTRODUCCIÓN..... | 5 |
| 2. DEFINICIÓN DEL PROBLEMA..... | 6 |
| 2.1. Contexto..... | 6 |
| 2.2. Problema..... | 6 |
| 2.3. Solución..... | 6 |
| 3. OBJETIVOS..... | 7 |
| 3.1. Objetivo General..... | 7 |
| 3.2. Objetivos Específicos..... | 7 |
| 3.3. Alcance..... | 7 |
| 4. PLANIFICACIÓN..... | 8 |
| 4.1. Metodología..... | 8 |
| 4.1.1 Marco de trabajo..... | 8 |
| 4.1.2 ¿Por qué Scrum?..... | 8 |
| 4.2. Mecanismos de Organización..... | 9 |
| 4.2.1 Estructura del Equipo..... | 9 |
| 4.2.2 Herramientas y Software..... | 9 |
| 4.3. Carta Gantt..... | 10 |
| 5. RESUMEN DEL PROYECTO..... | 11 |
| 5.1. Documento de Requisitos de Alto Nivel..... | 11 |
| 5.1.1. Características Principales..... | 11 |
| 5.1.1.1. Requerimientos Funcionales..... | 11 |
| 5.1.1.2. Requerimientos No Funcionales..... | 11 |
| 5.1.1.3. Restricciones..... | 12 |
| 5.1.1.4. Entregables..... | 12 |
| 5.1.2. Alcance del Ambiente del Software..... | 12 |
| 5.2. Diseño del Proyecto..... | 13 |
| 5.2.1. Arquitectura del Sistema..... | 13 |
| 5.2.2. Modelo de Contexto..... | 13 |
| 5.2.3. Identificación de Subsistemas..... | 14 |
| 5.2.4. Modelo de Casos de Uso..... | 15 |
| 5.2.5. Desarrollo de un BPM..... | 20 |
| 5.2.6. Herramientas para el Desarrollo del Software..... | 23 |
| 5.2.7. Modelamiento de Datos..... | 24 |
| 5.2.8. Aspectos Iniciales del Front-End..... | 27 |
| 6. CONCLUSIÓN..... | 33 |
| 7. REFERENCIAS..... | 34 |

ÍNDICE DE FIGURAS.

| | |
|--|----|
| Figura 1. Logo de la Facultad de Odontología de la Universidad de Chile..... | 5 |
| Figura 2. Carta Gantt del Proyecto..... | 10 |
| Figura 4. Arquitectura de Sistema Inicial..... | 13 |
| Figura 5. Diagrama de Modelo de Contexto..... | 13 |
| Figura 6. Diagrama de Modelo de Subsistema..... | 14 |
| Figura 7. Casos de Uso para “Servidor de Presupuesto”..... | 15 |
| Figura 8. Casos de Uso para “Administrar Usuario”..... | 15 |
| Figura 9. Casos de Uso para “Administrar Centro de Costos”..... | 16 |
| Figura 10. Casos de Uso para “Administrar Subcentro de Costos”..... | 16 |
| Figura 11. Casos de Uso para “Administrar Presupuesto Anual”..... | 17 |
| Figura 12. Casos de Uso para “Administrar Presupuesto para Centro de Costos”..... | 17 |
| Figura 13. Casos de Uso para “Administrar Presupuesto para Subcentro de Costos”..... | 18 |
| Figura 14. Casos de Uso para “Administrar Categorías”..... | 18 |
| Figura 15. Casos de Uso para “Administrar Transacción”..... | 19 |
| Figura 16. Casos de Uso para “Servidor de Autenticación”..... | 19 |
| Figura 17. Diagrama BPMN del Proceso “Gestionar Centros de Costo”..... | 20 |
| Figura 18. Diagrama BPMN del Proceso “Asignar Presupuesto”..... | 21 |
| Figura 19. Diagrama BPMN del Proceso “Gestionar Gastos”..... | 22 |
| Figura 20. Boceto de Pantalla “Inicio de Sesión”..... | 24 |
| Figura 21. Boceto de Pantalla “Inicio de Sesión”..... | 27 |
| Figura 22. Boceto de Pantalla Inicial del Sistema Administrativo..... | 27 |
| Figura 23. Boceto de Pantalla “Gestión de Centros de Costo”..... | 28 |
| Figura 24. Boceto de Pantalla “Gestión de Subcentros de Costo”..... | 28 |
| Figura 25. Boceto de Pantalla “Gestión de Usuarios”..... | 29 |
| Figura 26. Boceto de Pantalla “Creación de Usuario”..... | 29 |
| Figura 27. Boceto de Pantalla “Gestión de Presupuesto General Anual”..... | 30 |
| Figura 28. Boceto de Pantalla “Gestión de Presupuesto de Centros de Costo”..... | 30 |
| Figura 29. Boceto de Pantalla “Asignación de Presupuesto de Centro de Costo”..... | 31 |
| Figura 30. Boceto de Pantalla “Panel de Responsable de Centro de Costo”..... | 31 |
| Figura 31. Boceto de Pantalla “Crear Compra”..... | 32 |

ÍNDICE DE TABLAS.

| | |
|---|----|
| Tabla 1. Estructura del Equipo..... | 9 |
| Tabla 2. Herramientas y Software..... | 9 |
| Tabla 3. Requerimientos Funcionales..... | 11 |
| Tabla 4. Requerimientos no Funcionales..... | 11 |

1. INTRODUCCIÓN

La gestión eficiente de los recursos financieros es un aspecto fundamental para el funcionamiento óptimo de cualquier institución educativa. En el contexto de la Facultad de Odontología de la Universidad de Chile [1], se enfrentan desafíos significativos en la asignación y control de presupuestos destinados a diferentes áreas académicas y clínicas. La complejidad aumenta al tener que distribuir un presupuesto general entre múltiples centros de costos y mantener un registro preciso de los gastos realizados.

El presente proyecto tiene como objetivo desarrollar un sistema de gestión y control de presupuesto web que permita gestionar de manera eficiente y transparente los recursos financieros de la Facultad de Odontología. Esta solución integrará tecnologías modernas como la autenticación con Google, un sistema de mensajería para comunicaciones internas, y funcionalidades que faciliten la asignación y seguimiento de presupuestos en tiempo real.

A través de este proyecto, se busca mejorar la toma de decisiones basadas en información actualizada y precisa, optimizar los procesos de asignación y reducción de presupuestos, y garantizar la transparencia y eficiencia en el uso de los recursos financieros disponibles.



Figura 1. Logo de la Facultad de Odontología de la Universidad de Chile.

2. DEFINICIÓN DEL PROBLEMA

2.1. Contexto

El proyecto a desarrollar tiene como objetivo abordar una problemática de control de presupuesto dentro de la Facultad de Odontología de la Universidad de Chile. La Clínica Odontológica de la Universidad de Chile es una institución de salud que cuenta con 148 unidades dentales y ofrece atención de problemas odontológicos y tratamiento de enfermedades bucales.

2.2. Problema

En la actualidad, la gestión de presupuestos en la Facultad de Odontología se lleva a cabo de forma manual y descentralizada, lo que complica la asignación eficiente de recursos y el seguimiento de gastos. La ausencia de un sistema centralizado y automatizado ocasiona retrasos en los procesos y dificulta la toma de decisiones fundamentadas en información actualizada y precisa.

2.3. Solución

Para abordar esta problemática, se propone el desarrollo e implementación de un sistema de gestión y control de presupuesto a través de una plataforma web. Esta solución permitirá optimizar la asignación y distribución de recursos financieros en la Facultad de Odontología. Entre las características principales de esta solución se encuentra la creación de perfiles de usuario con diferentes niveles de acceso, funcionalidades para gestionar el presupuesto de los distintos centros de costos de la Clínica Odontológica de la Universidad de Chile y para visualizar detalladamente los gastos producidos por cada centro de costo.

3. OBJETIVOS

3.1. Objetivo General

Desarrollar un sistema de gestión y control de presupuesto para la Facultad de Odontología de la Universidad de Chile que permita asignar, distribuir y controlar los recursos financieros de manera eficiente y transparente.

3.2. Objetivos Específicos

- Estudiar y analizar el ambiente de trabajo de la Facultad de Odontología para identificar la problemática en el proceso de gestión presupuestario.
- Analizar y definir el alcance del sistema para establecer los requerimientos funcionales y no funcionales necesarios para llevar a cabo el proyecto.
- Implementar el sistema de gestión y control de presupuesto para permitir una gestión eficiente de los recursos.
- Realizar las pruebas de funcionamiento y análisis de resultados para asegurar que el sistema cumple con los requerimientos establecidos.

3.3. Alcance

El sistema abarca la asignación y distribución de presupuestos anuales para áreas académicas y clínicas específicas dentro de la Facultad de Odontología. Se incluirá la funcionalidad para reducir presupuestos en centros de costos mediante compras o contrataciones. Además, se desarrollará un módulo de control de acceso y autenticación para garantizar la seguridad de la plataforma.

4. PLANIFICACIÓN

4.1. Metodología

4.1.1 Marco de trabajo

El marco de trabajo a utilizar en este proyecto es Scrum. Scrum es un marco de trabajo basado en metodologías ágiles para la gestión de proyectos, enfocados en un conjunto de valores, principios y prácticas. Scrum propone un enfoque iterativo e incremental que se ajusta perfectamente a proyectos que requieren adaptabilidad y respuesta rápida a los cambios.

4.1.2 ¿Por qué Scrum?

Se escogió Scrum ya que este marco de trabajo impulsa la adaptabilidad, la flexibilidad, la comunicación y la retroalimentación. Estos pilares son de suma importancia al momento de entablar acuerdos con el cliente alcanzando un mayor conocimiento y apoyo mutuo en los requisitos que se deben cumplir en el proyecto, gracias a la gran comunicación y retroalimentación que este marco de trabajo propone.

4.2. Mecanismos de Organización

4.2.1 Estructura del Equipo

Tabla 1. Estructura del Equipo.

| Rol | Responsabilidad | Responsable |
|-------------------|--|----------------|
| Jefe del Proyecto | <ul style="list-style-type: none"> • Coordinación general del proyecto, asegurando el cumplimiento de plazos y objetivos. • Gestión de recursos y asignación de tareas. • Monitoreo del progreso del proyecto y ajustes de planificación según sea necesario. | Diego Baltazar |
| Desarrollador | <ul style="list-style-type: none"> • Participar en la definición de los requisitos del software junto con el equipo y los usuarios. • Diseñar la arquitectura del sistema y el diseño de interfaces. • Desarrollar el software siguiendo las mejores prácticas de desarrollo. | Dylan Choque |

4.2.2 Herramientas y Software

Tabla 2. Herramientas y Software.

| Nombre | Logo | Funcion |
|--------------------|---|--|
| Whatsapp |  | Comunicación rápida y efectiva entre miembros del equipo. |
| Notion |  | Organización de la documentación del proyecto, notas, y tareas pendientes. |
| Drive |  | Almacenamiento de documentos y archivos importantes del proyecto en la nube. |
| Draw.io |  | Herramienta de diagramación para la creación de diagramas arquitectónicos, flujos de trabajo y otros diagramas relacionados con el proyecto. |
| Visual Studio Code |  | Editor de código fuente, utilizado para la programación y desarrollo del proyecto. |
| Github |  | Control de versiones y almacenamiento del código fuente del proyecto. |
| Redmine |  | Gestión de proyectos y seguimiento de errores; subida y evaluación de documentos. |



4.3. Carta Gantt

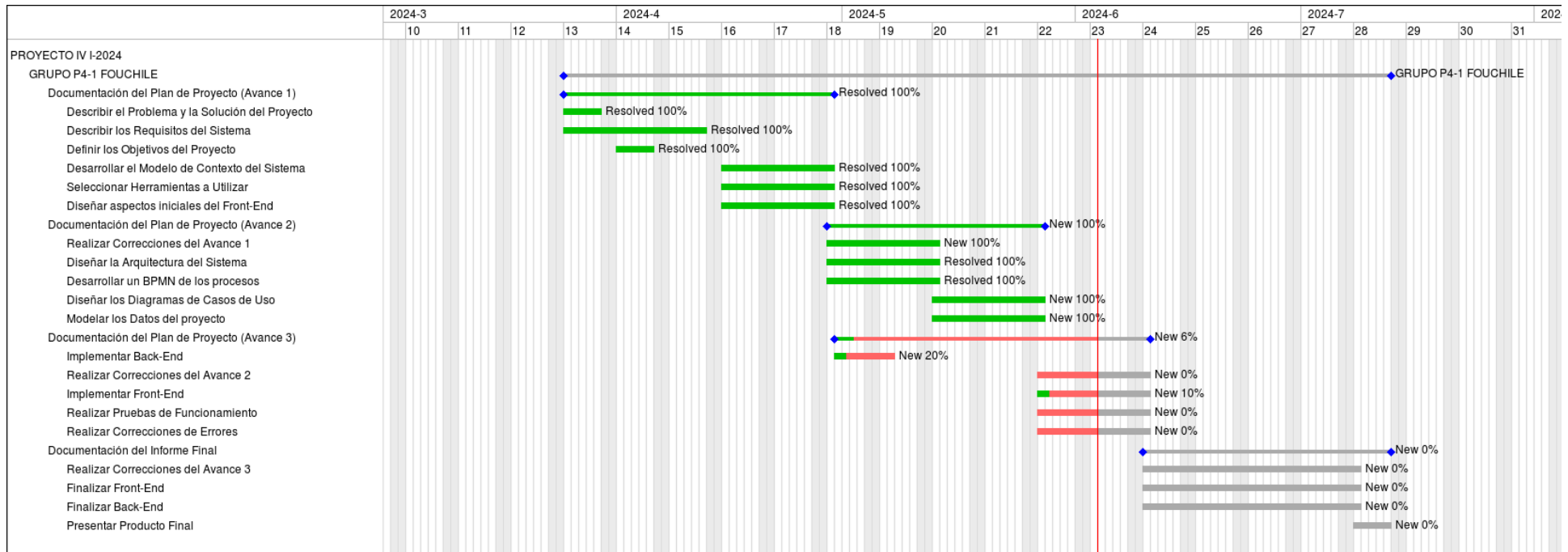


Figura 2. Carta Gantt del Proyecto.

5. RESUMEN DEL PROYECTO

5.1. Documento de Requisitos de Alto Nivel

5.1.1. Características Principales

El sistema de control de presupuesto de la Facultad de Odontología incluirá las siguientes características principales:

5.1.1.1. Requerimientos Funcionales.

Tabla 3. Requerimientos Funcionales.

| NºRequerimiento | Descripción |
|-----------------|---|
| RF_1 | Los usuarios deben poder iniciar sesión según su rol en el sistema utilizando credenciales válidas. |
| RF_2 | Permitir a los administradores asignar y modificar los presupuestos de cada uno de los centros y subcentros de costos de la facultad. |
| RF_3 | Asignar y modificar usuarios responsables para cada centro de costos de la facultad. |
| RF_4 | Visualizar la ramificación de los gastos realizados por cada centro de costos, incluyendo los ítems específicos en los que se ha gastado. |
| RF_5 | Generar alertas automáticas para los responsables de los centros de costos cuando el saldo disponible se encuentre en un porcentaje bajo. |
| RF_6 | Generar informes financieros por periodos de los gastos producidos en distintos centros de costos de la facultad. |

5.1.1.2. Requerimientos No Funcionales.

Tabla 4. Requerimientos no Funcionales.

| NºRequerimiento | Descripción |
|-----------------|--|
| RnF_1 | El sistema debe cumplir con los estándares de seguridad requeridos para mantener la integridad y seguridad de los datos. |
| RnF_2 | Interfaz interactiva y amigable con el usuario para facilitar la navegación y gestión dentro del sistema. |
| RnF_3 | Los usuarios del sistema deben poder acceder mediante navegador web en computadores y dispositivos móviles. |
| RnF_4 | El sistema debe adoptar una arquitectura de microservicios y ser capaz de comunicarse con sistemas externos. |
| RnF_5 | El código debe estar bien documentado y estructurado para facilitar futuras actualizaciones y mantenimiento. |

5.1.1.3. Restricciones.

- Plataforma web: El sistema estará disponible como una aplicación web accesible desde navegadores modernos.
- Integridad de datos: Implementar mecanismos de control de versiones y copias de seguridad periódicas para mantener la integridad de los datos y la capacidad de recuperación en caso de fallos o pérdida de información.
- Escalabilidad: Diseñar el sistema de manera que pueda escalar fácilmente para manejar un aumento en el número de usuarios, centros de costos y transacciones, sin comprometer el rendimiento ni la seguridad.

5.1.1.4. Entregables.

- Bitácoras Semanales
- Documentos de Requerimientos Firmado por el Cliente
- Informe de Avance 1
- Informe de Avance 2
- Informe de Avance 3
- Informe Final
- Manuel de Usuario
- Propuesta de Despliegamiento
- Documentación del Producto
- Producto

5.1.2. Alcance del Ambiente del Software

El alcance del ambiente del software incluirá la implementación de todas las funcionalidades mencionadas anteriormente en un entorno web seguro y escalable. Se desarrollarán módulos específicos para la gestión de presupuestos, autenticación y otras características esenciales para el funcionamiento del sistema.

5.2. Diseño del Proyecto

5.2.1. Arquitectura del Sistema

La arquitectura del sistema propuesto se basa en un enfoque de microservicios utilizando Angular para el frontend y NestJS para los servicios backend, organizados en dos microservicios principales: uno para la gestión del presupuesto y otro para la autenticación. La comunicación entre los microservicios se facilita mediante TCP, asegurando un flujo de datos eficiente y desacoplado. Cada servicio cuenta con su propia base de datos MongoDB. Todo el sistema está contenerizado con Docker, permitiendo una implementación y mantenimiento flexibles y consistentes en diferentes entornos.

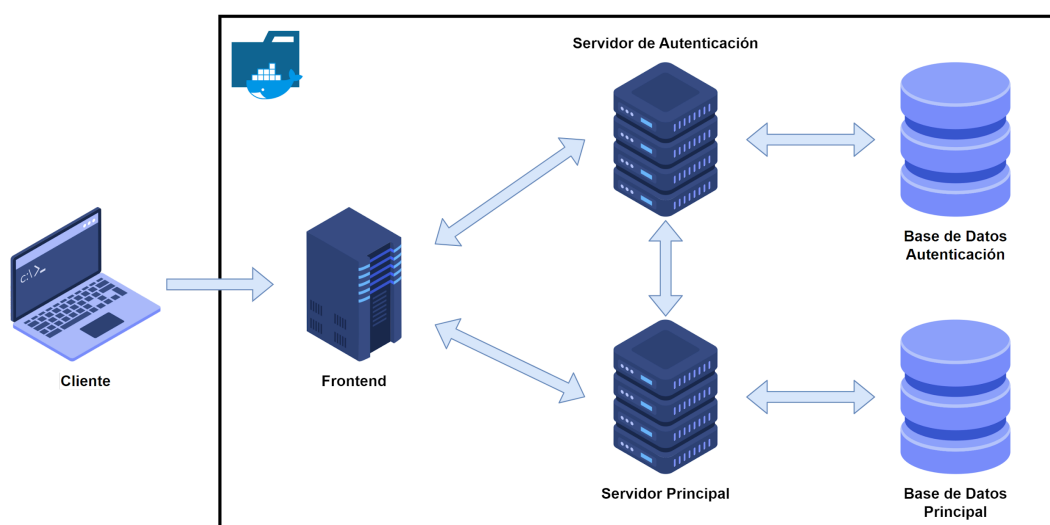


Figura 4. Arquitectura de Sistema Inicial.

5.2.2. Modelo de Contexto

El modelo de contexto ilustrado en la figura 5 muestra cómo el sistema de gestión y control de presupuesto interactúa con los usuarios dentro de la Facultad de Odontología.

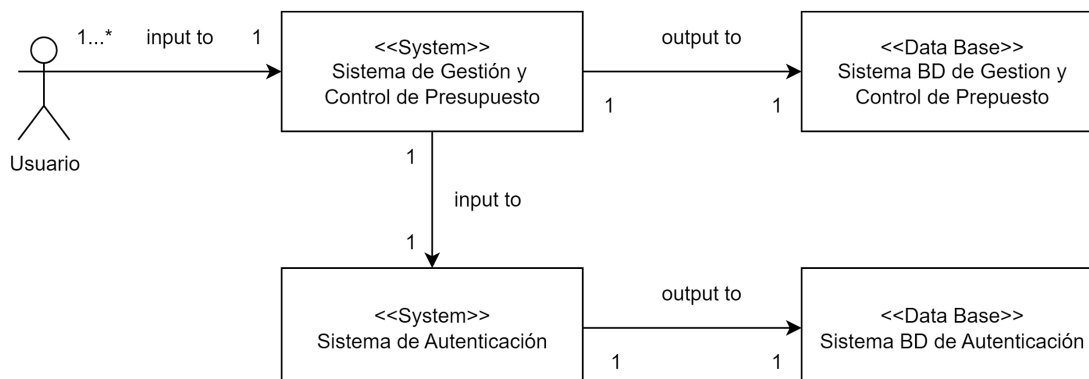


Figura 5. Diagrama de Modelo de Contexto.

5.2.3. Identificación de Subsistemas

El diagrama de subsistemas ilustrado en la figura 6 muestra la estructura y las interacciones del sistema de gestión y control de presupuestos de la Facultad de Odontología, podemos organizar los subsistemas basándonos en las principales funcionalidades y responsabilidades.

- Subsistema de Interfaz Web: Este módulo es responsable de interactuar con los usuarios finales y comunicarse con los microservicios de gestión de presupuesto y autenticación para obtener y enviar datos.
- Subsistema de Gestión y Control de Presupuesto: Este módulo es responsable del manejo de los presupuestos, encargándose de las operaciones relacionadas con la asignación, modificación y seguimiento de los fondos a través de los centros de costos.
- Subsistema de Base de Datos del Sistema de Gestión y Control de Presupuesto: Este módulo es responsable de almacenar y gestionar todos los datos, incluyendo registros de transacciones y ajustes de fondos.
- Sistema de Autenticación: Este módulo es responsable de la gestión de la autenticación y autorización de los usuarios que acceden al sistema. Este módulo asegura que solo los usuarios autorizados puedan realizar operaciones dentro del sistema de gestión presupuestaria.
- Subsistema de Base de Datos del Sistema de Autenticación: Este módulo es responsable de proporcionar una gestión segura de las credenciales y los roles de acceso.

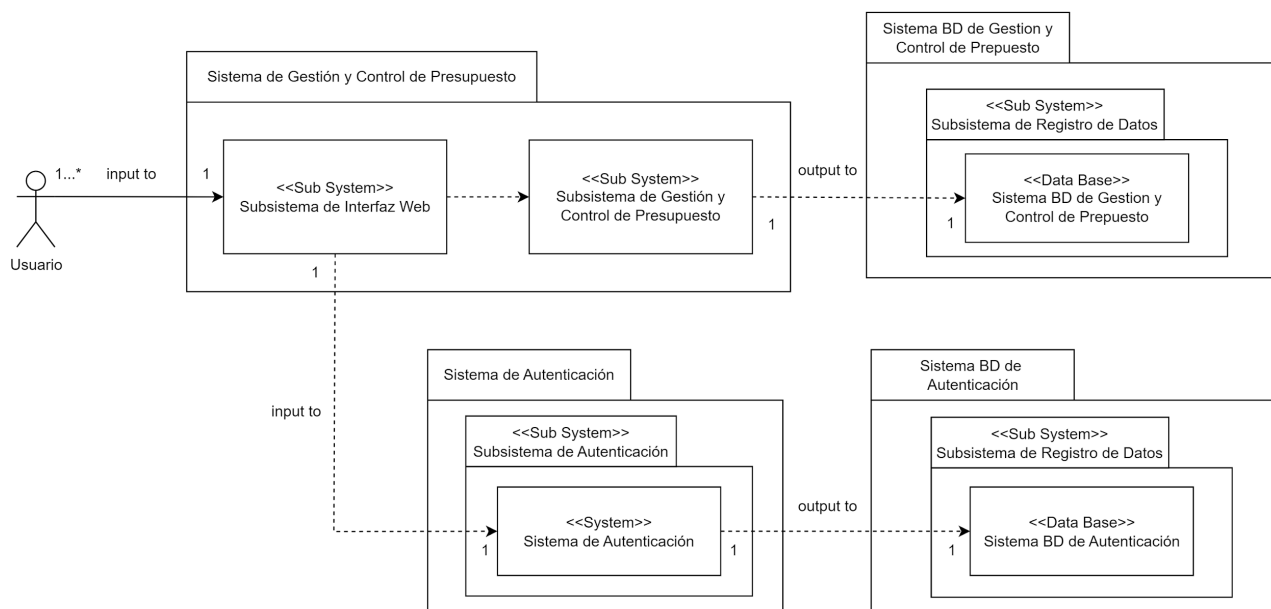


Figura 6. Diagrama de Modelo de Subsistema.

5.2.4. Modelo de Casos de Uso

A continuación, se presentan descripciones iniciales para los diagramas de casos de uso específicos de diferentes componentes del sistema de gestión y control de presupuestos:

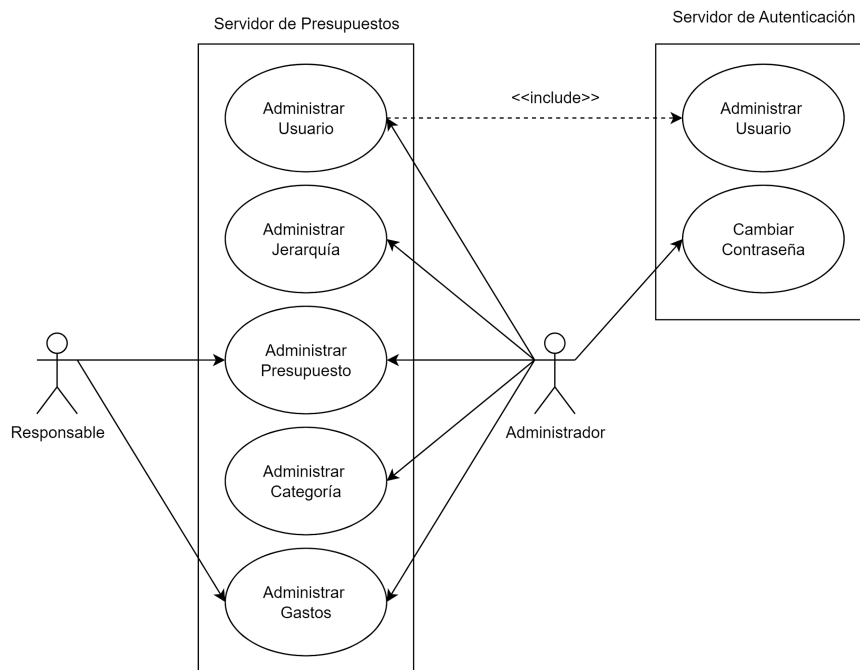


Figura 7. Casos de Uso para “Servidor de Presupuesto”.

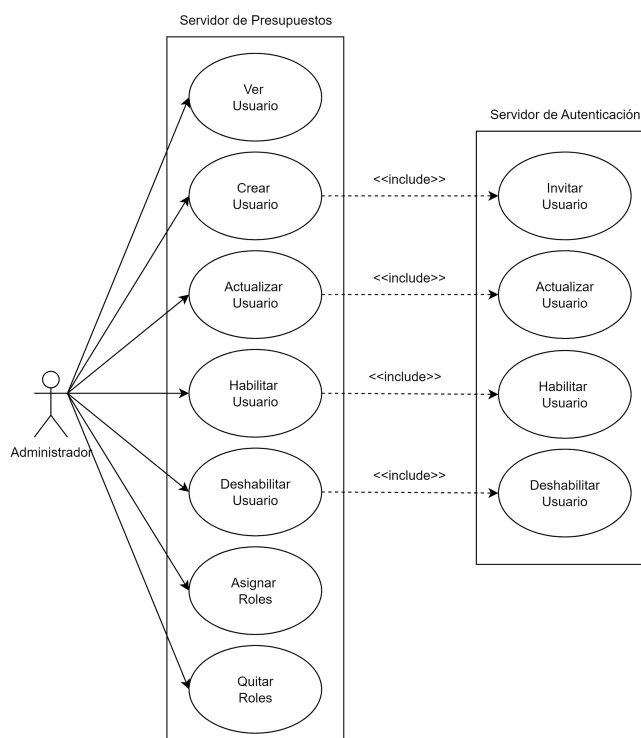


Figura 8. Casos de Uso para “Administrar Usuario”.

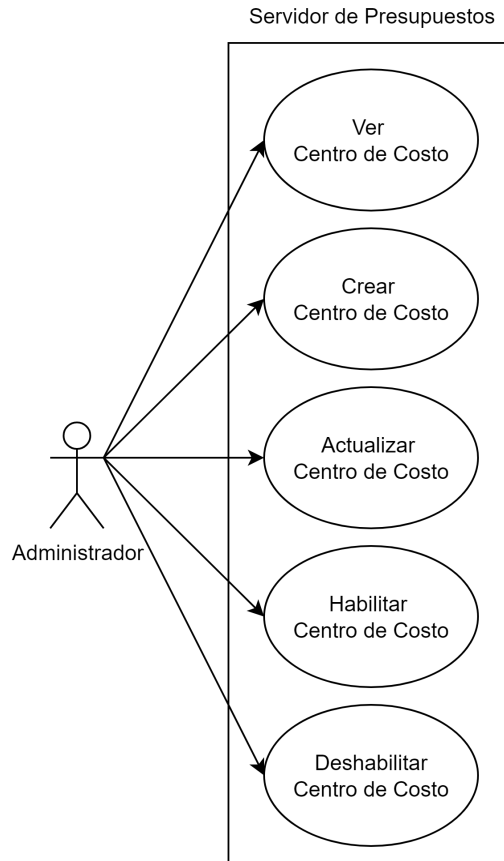


Figura 9. Casos de Uso para “Administrar Centro de Costos”.

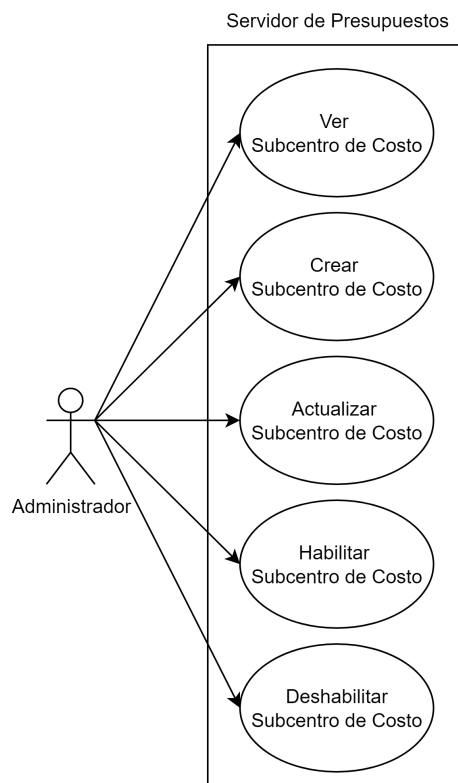


Figura 10. Casos de Uso para “Administrar Subcentro de Costos”.

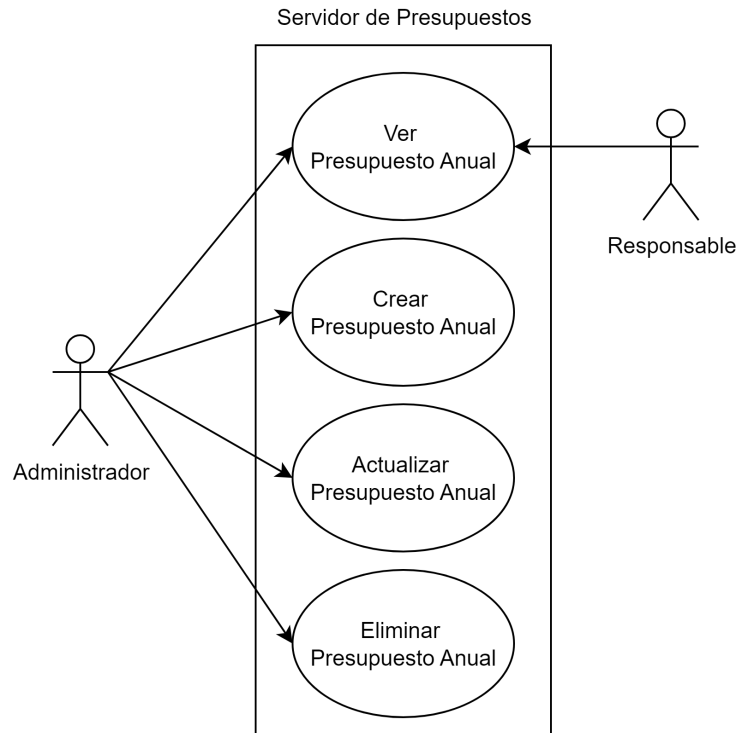


Figura 11. Casos de Uso para “Administrar Presupuesto Anual”.

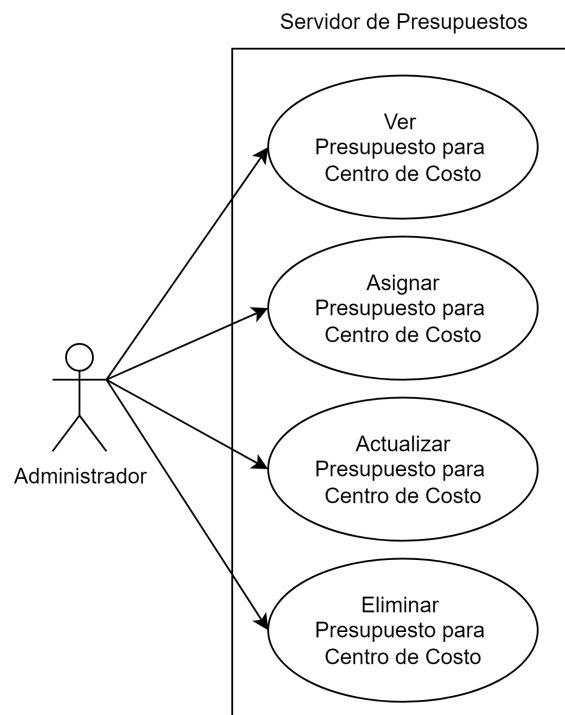


Figura 12. Casos de Uso para “Administrar Presupuesto para Centro de Costos”.

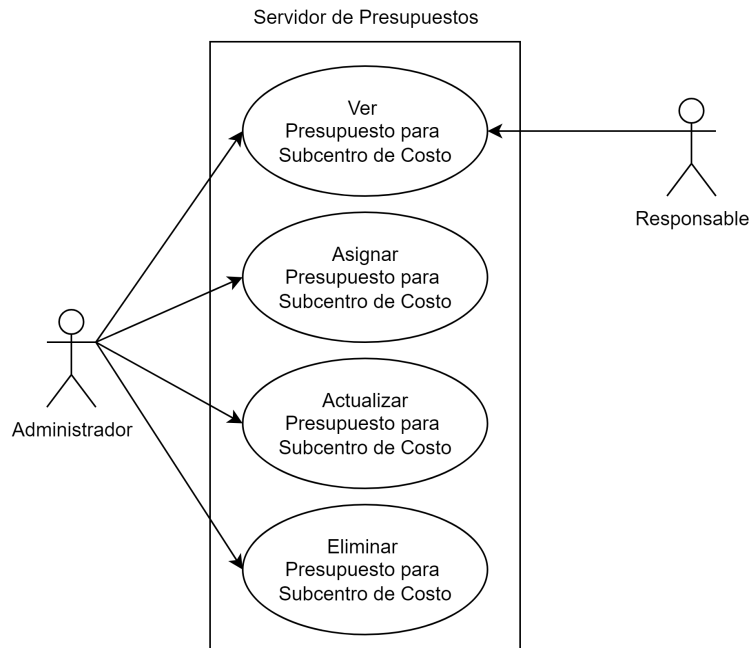


Figura 13. Casos de Uso para “Administrar Presupuesto para Subcentro de Costos”.

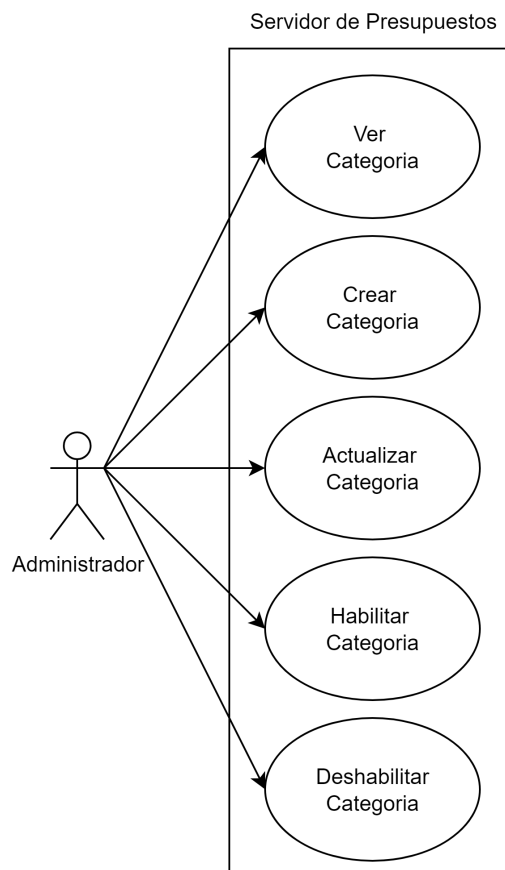


Figura 14. Casos de Uso para “Administrar Categorías”.

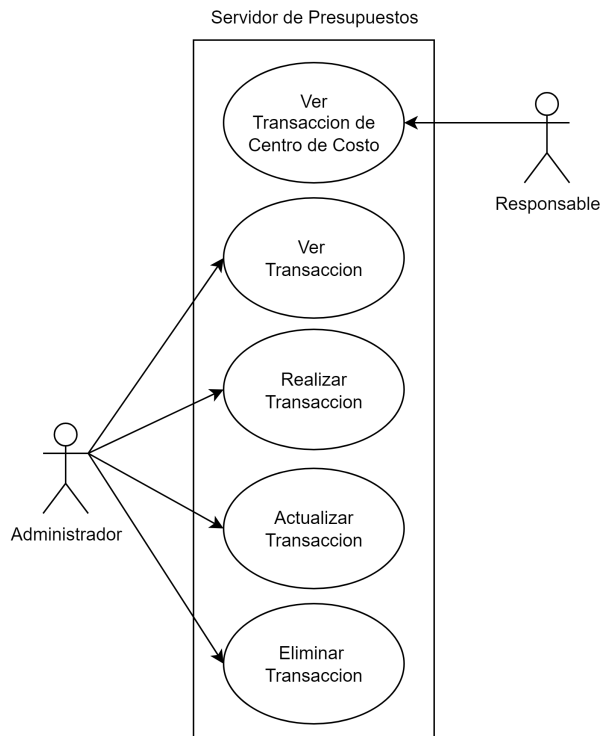


Figura 15. Casos de Uso para "Administrar Transacción".

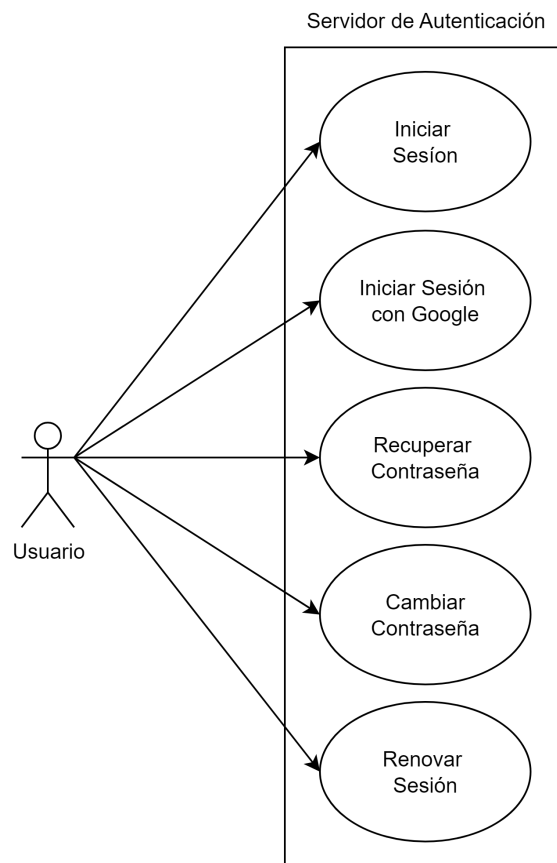


Figura 16. Casos de Uso para "Servidor de Autenticación".

5.2.5. Desarrollo de un BPM.

El Modelado de Procesos de Negocio (BPM) es un método de diagrama de flujo que modela los pasos de un proceso de negocio planificado desde un inicio a un fin. Es clave para la gestión de procesos de negocio y representa visualmente una secuencia detallada de actividades comerciales y flujos de información necesarios para completar un proceso. Se utilizó la herramienta Bizagi Modeler [3] para el modelado de los distintos procesos para la Gestión de Presupuesto de la Facultad de Odontología mediante el sistema a implementar.

En la Figura 17 se visualiza el proceso de “Gestión de Centros de Costo” el cuál consiste en la creación de los distintos centros y subcentros de costo de la Facultad de Odontología. Al inicio el administrador crea los Centros de Costo, una vez creados, debe crear los subcentros de costo de cada centro. Finalmente asigna a una persona responsable para cada centro y subcentro de costo.

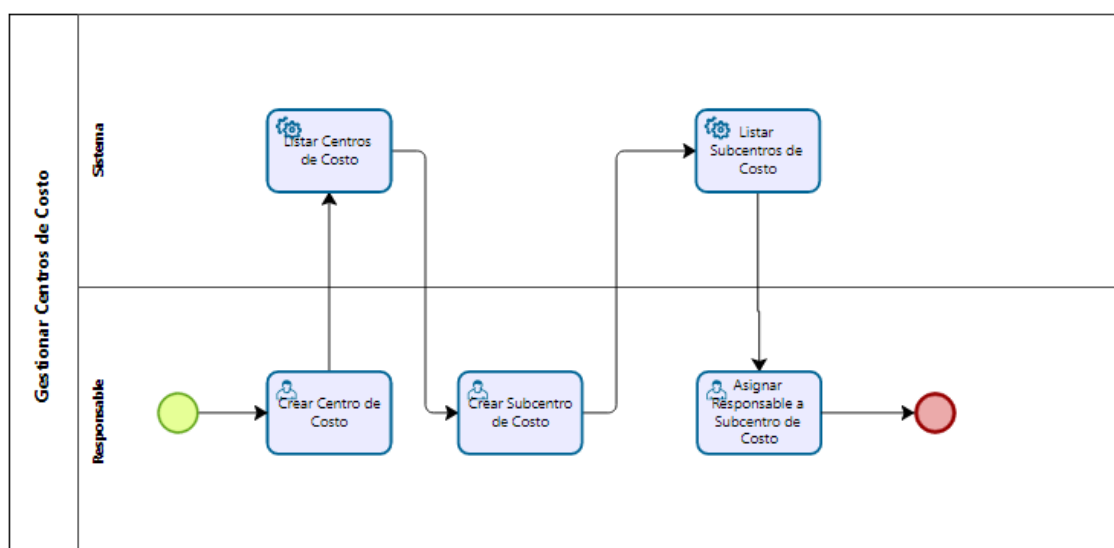


Figura 17. Diagrama BPMN del Proceso “Gestionar Centros de Costo”.

En la Figura 18 se visualiza el proceso de “Asignar Presupuesto” el cuál consiste en la asignación de presupuesto a los distintos centros y subcentros de costo de la Facultad de Odontología. Al inicio el administrador debe asignar un Presupuesto General el cuál corresponde al presupuesto anual con el que cuenta toda la Facultad, una vez asignado, debe asignar el presupuesto de los centros de costo los cuáles serán descontados del Presupuesto General, luego debe asignar el presupuesto de cada subcentro el cuál será descontado del presupuesto del centro al que pertenece. Finalmente la persona responsable de los centros de costo, podrá visualizar el presupuesto de su centro.

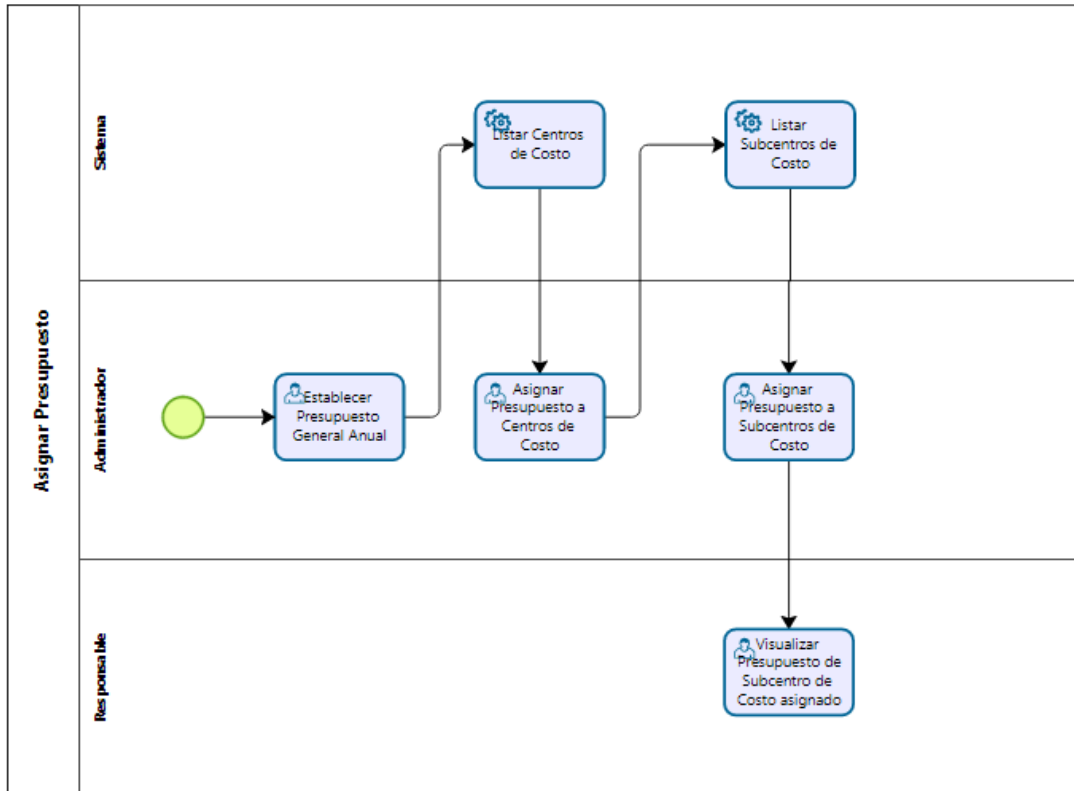


Figura 18. Diagrama BPMN del Proceso "Asignar Presupuesto".

En la Figura 19 se visualiza el proceso de "Gestionar Gastos" el cuál consiste en la gestión de información de los gastos hechos por la Facultad de Odontología. Al inicio el administrador debe ingresar los detalles de un gasto realizado especificando el monto y el subcentro de costo que realizó el gasto. Luego el monto será descontado del presupuesto del subcentro de costo que se ingresó en los detalles del gasto. Finalmente tanto el administrador como las personas responsables podrán visualizar los gastos realizados por los distintos centros de costo de la Facultad.

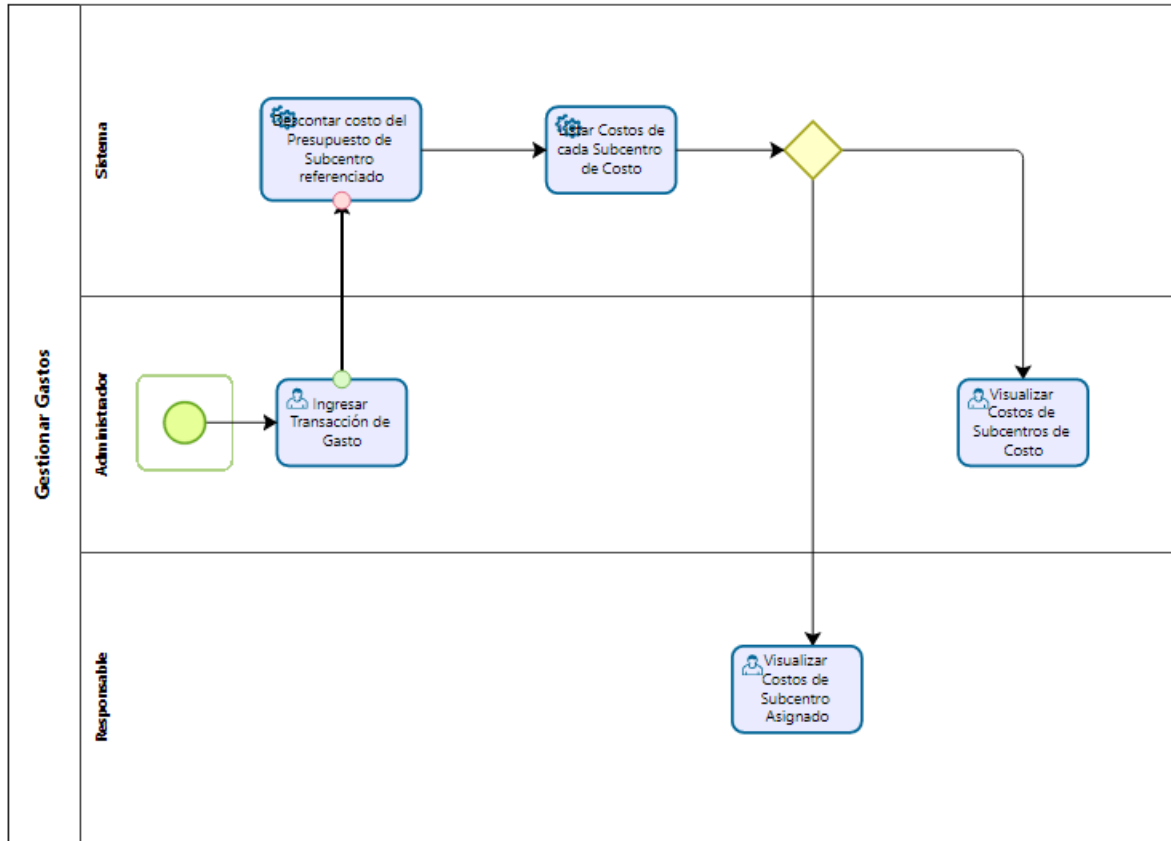


Figura 19. Diagrama BPMN del Proceso “Gestionar Gastos”.

5.2.6. Herramientas para el Desarrollo del Software

1. Angular:
 - Función: Framework de desarrollo para construir la interfaz de usuario del sistema. Facilita la creación de componentes reutilizables y maneja eficientemente el estado y la lógica de la interfaz.
2. NestJS:
 - Función: Framework de Node.js que se utiliza para construir aplicaciones del lado del servidor. Es ideal para la creación de servicios robustos y escalables, apoyando el modelo de microservicios con su arquitectura modular y facilidad de integración con otros sistemas y bases de datos [\[4\]](#).
3. MongoDB:
 - Función: Base de datos NoSQL utilizada para almacenar y recuperar datos de manera eficiente. Su esquema flexible es ideal para manejar las estructuras de datos complejas requeridas por los microservicios [\[5\]](#).
4. Docker:
 - Función: Plataforma de contenerización que facilita el desarrollo, el testing y la producción de aplicaciones al permitir que los microservicios se empaqueten y se ejecuten en contenedores independientes. Esto asegura consistencia entre los entornos de desarrollo, prueba y producción [\[6\]](#).
5. TCP:
 - Función: Middleware de mensajería por defecto de Nest JS que permite la comunicación entre diferentes microservicios a través de la transmisión de mensajes de manera confiable y eficiente, lo cual es crucial en una arquitectura de microservicios [\[7\]](#).
6. Visual Studio Code:
 - Función: Editor de código que soporta múltiples lenguajes de programación y tiene extensiones para integrar otros servicios y herramientas como Docker y Git, lo que lo hace ideal para el desarrollo de software moderno.
7. Git (con GitHub):
 - Función: Sistema de control de versiones que permite gestionar el código fuente y colaborar en el desarrollo del proyecto. GitHub proporciona repositorios remotos, revisión de código, gestión de contribuciones y más.

5.2.7. Modelamiento de Datos.

El modelamiento de datos para el sistema de gestión y control de presupuesto se ha realizado utilizando draw.io [9] para el diseño del diagrama, y la base de datos como se especificó anteriormente, es MongoDB. A continuación en la Figura 20 se presenta el modelo de Base de Datos del sistema.

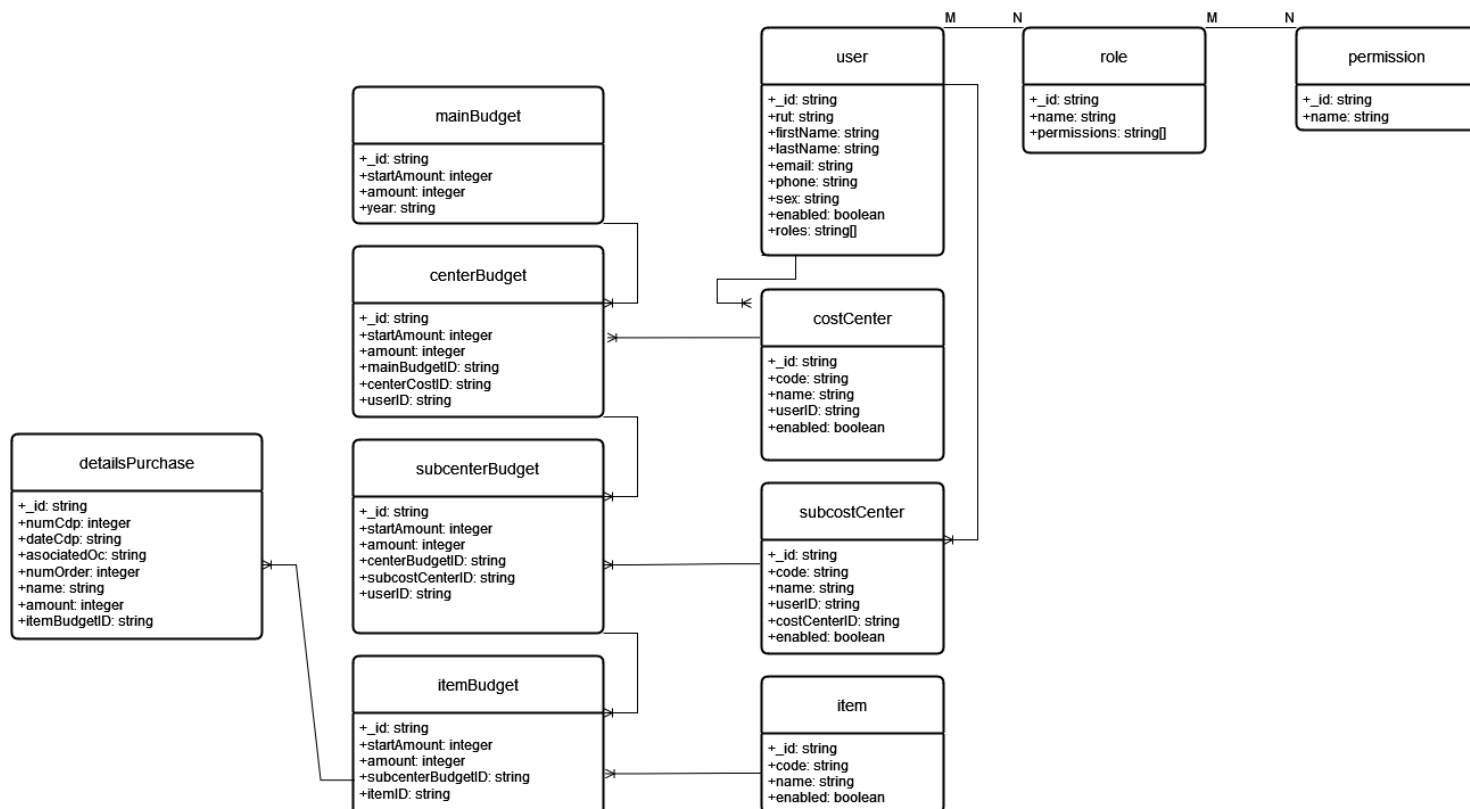


Figura 20. Boceto de Pantalla “Inicio de Sesión”.

A continuación se explicarán las entidades y relaciones que contiene el modelo de datos:

User: Corresponde a los usuarios que pueden ser de rol Administrador o Responsable, este cuenta con los atributos: rut, firstName, lastName, email, phone, sex y roles[].

Entre sus relaciones tenemos que User posee N:M Roles, un User tiene varios centros de costo pero un centro de costo solo puede tener un User asignado y un User tiene varios subcentros de costo pero un subcentro de costo solo puede tener un User asignado.

Role: Corresponde a los roles que tendrá el sistema, este cuenta con los atributos: name y permissions[].

Entre sus relaciones tenemos que Role posee N:M Permissions y un Role posee N:M Users.

Permission: Corresponde a los permisos que tendrá el sistema, este cuenta con los atributos: name.

Entre sus relaciones tenemos que Permission posee N:M Roles.

CostCenter: Corresponde a los centros de costo de la Facultad de Odontología, este cuenta con los atributos: code, name, userID, enabled.

Entre sus relaciones tenemos que un costCenter posee un User asignado pero un User puede tener varios costCenter asignados, un costCenter puede tener varios subcostCenter pero un subcostCenter posee un solo costCenter y que un costCenter puede tener varios centerBudgets pero un centerBudget sólo puede tener asignado un costCenter.

SubcostCenter: Corresponde a los subcentros de costo de la Facultad de Odontología, este cuenta con los atributos: code, name, userID, costCenterID, enabled.

Entre sus relaciones tenemos que un subcostCenter posee un User asignado pero un User puede tener varios subcostCenter asignados, un subcostCenter sólo puede tener un costCenter pero un costCenter puede tener varios subcostCenter y que un subcostCenter puede tener varios subcenterBudgets pero un subcenterBudget sólo puede tener asignado un subcostCenter.

Ítem: Corresponde a los ítems presupuestarios que se usan para categorizar los gastos, este cuenta con los atributos: code, name, userID, enabled.

Entre sus relaciones tenemos que un Ítem solo tiene un subcostCenter pero un subcostCenter puede tener varios Ítem y un Ítem puede tener varios itemBudgets pero un itemBudget sólo puede tener asignado un Ítem.

MainBudget: Corresponde a los presupuestos generales de cada año de la Facultad de Odontología, este cuenta con los atributos: startAmount, amount, year.

Entre sus relaciones tenemos que un mainBudget puede tener varios costcenterBudgets pero un costcenterBudget sólo puede tener asignado un mainBudget.

CenterBudget: Corresponde a los centros de costo de la Facultad de Odontología, este cuenta con los atributos: startAmount, amount, mainBudgetID, costCenterID, UserID.

Entre sus relaciones tenemos que un centerBudget solo tiene un mainBudget pero un mainBudget puede tener varios centerBudgets y un costCenter puede tener varios centerBudgets pero un centerBudget sólo puede tener asignado un costCenter.

SubcenterBudget: Corresponde a los presupuestos de los subcentros de costo de la Facultad de Odontología, este cuenta con los atributos: startAmount, amount, mainBudgetID, subcostCenterID, UserID.

Entre sus relaciones tenemos que un subcenterBudget solo tiene un centerBudget pero un centerBudget puede tener varios subcenterBudgets y un subcenterBudget puede tener asignado un solo subcostCenter pero un subcostCenter puede tener varios subcenterBudgets.

itemBudget: Corresponde al presupuesto que posee un Ítem en el año, este cuenta con los atributos: startAmount, amount, subcenterBudgetID, itemID .

Entre sus relaciones tenemos que un itemBudget solo tiene un subcenterBudget pero un subcenterBudget puede tener varios ÍtemBudget y un ÍtemBudget puede tener asignado un solo Ítem pero un Ítem puede tener varios ItemBudgets.

DetailsPurchase: Corresponde a los las compras realizadas por la Facultad de Odontología, este cuenta con los atributos: numCDP, dateCDP, asociatedOC, numOrder, name, amount, itemBudgetID.

Entre sus relaciones tenemos que un detailsPurchase solo tiene un Ítem asignado pero un Ítem puede tener varios detailsPurchase.

5.2.8. Aspectos Iniciales del Front-End.

Para el diseño de los bocetos iniciales del Frontend del Sistema, se utilizó la herramienta Balsamiq Wireframes [8]. A continuación se presentan los bocetos de las distintas pantallas que tendrá el sistema.



Figura 21. Boceto de Pantalla "Inicio de Sesión".

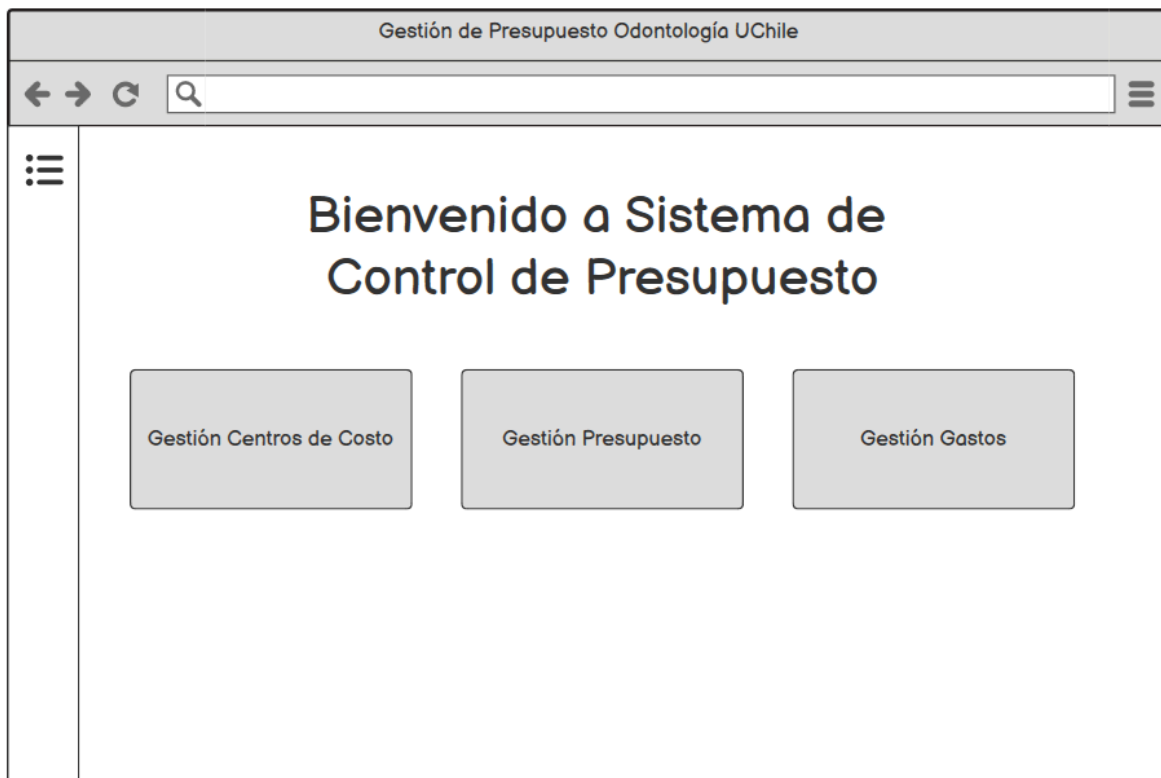


Figura 22. Boceto de Pantalla Inicial del Sistema Administrativo.

En la pantalla “Gestión de Centros de Costos” el administrador podrá crear y modificar los distintos Centros y Subcentros de Costo de la Facultad de Odontología de la Universidad de Chile, además podrá deshabilitarlos o asignar a un responsable.

Bienvenido a Gestión de Centro de Costos

Sistema Administrativo

Crear Centro

| CC | Nombre | Responsable | Acciones |
|--------|----------------------|----------------|---|
| 0100 | DECANATO | Diego Baltazar | Habilitado Modificar Subcentros Asignar Resp. |
| 0200 | VICEDECANATO | No Asignado | Habilitado Modificar Subcentros Asignar Resp. |
| 0300 | ESCUELA DE PREGRADO | No Asignado | Habilitado Modificar Subcentros Asignar Resp. |
| 0304 | DAA | No Asignado | Habilitado Modificar Subcentros Asignar Resp. |
| 0305 | DAE | No Asignado | Habilitado Modificar Subcentros Asignar Resp. |
| 3301 | DIFO | No Asignado | Habilitado Modificar Subcentros Asignar Resp. |
| 0400 | ESCUELA DE GRADUADOS | No Asignado | Habilitado Modificar Subcentros Asignar Resp. |
| 315900 | CLINICA DE PROYECTOS | No Asignado | Habilitado Modificar Subcentros Asignar Resp. |

Figura 23. Boceto de Pantalla “Gestión de Centros de Costo”.

DECANATO

Subcentros de Costo

Sistema Administrativo

Crear Subcentro

| CC | Nombre | Acciones |
|---------|------------------------------------|---------------------------------------|
| 0101001 | DIRECCION DECANATO | Habilitado Modificar Asignar Resp. |
| 0101005 | CURSO CLINICA (INYECCIÓN... | Habilitado Modificar Asignar Resp. |
| 0101006 | CURSO CLINICA REGULACIÓN... | Habilitado Modificar Asignar Resp. |
| 0101007 | COMITÉ DE ÉTICA CIENTÍFICO | Habilitado Modificar Asignar Resp. |
| 0101008 | CONTINGENCIA NACIONAL (PROTESIS... | Habilitado Modificar Asignar Resp. |
| 0101009 | PROYECTO COVID VAEGI | Habilitado Modificar Asignar Resp. |
| 0101010 | Estacionamiento de Quimica | Habilitado Modificar Asignar Resp. |
| 0101011 | PROGRAMA PEEI 2017-2019 | Deshabilitado Modificar Asignar Resp. |

Figura 24. Boceto de Pantalla “Gestión de Subcentros de Costo”.

En la pantalla “Gestión de Usuarios”, el Administrador podrá crear a los usuarios que podrán acceder al sistema, los usuarios pueden ser “Administrador” o “Responsable”.



Figura 25. Boceto de Pantalla “Gestión de Usuarios”.

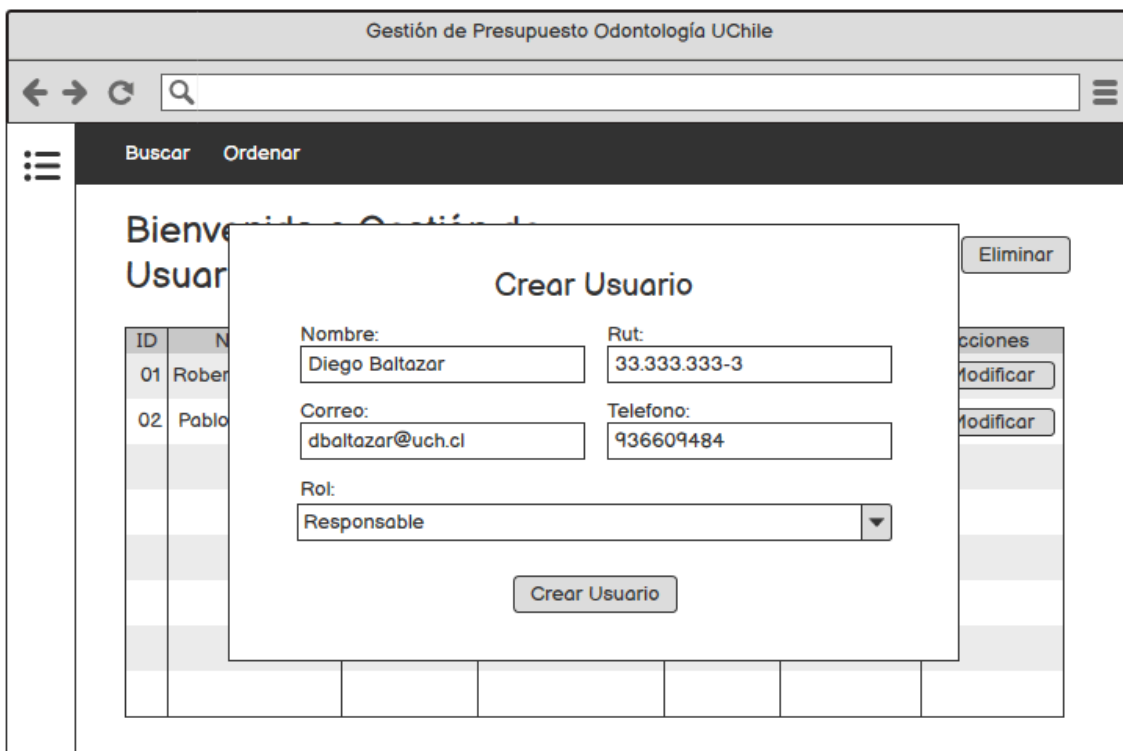


Figura 26. Boceto de Pantalla “Creación de Usuario”.

En la pantalla “Gestión de Presupuesto”, el Administrador podrá asignar el Presupuesto General Anual de la Facultad de Odontología, además podrá asignar también el Presupuesto para los Centros y Subcentros de la Facultad.

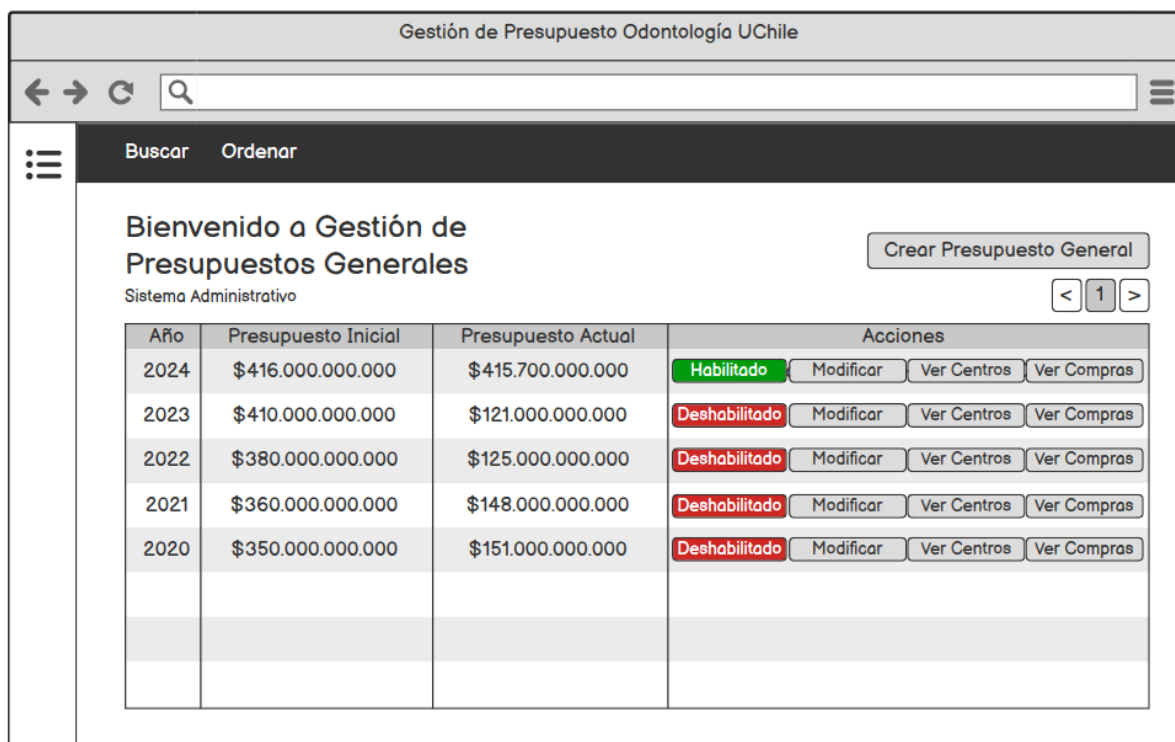


Figura 27. Boceto de Pantalla “Gestión de Presupuesto General Anual”.

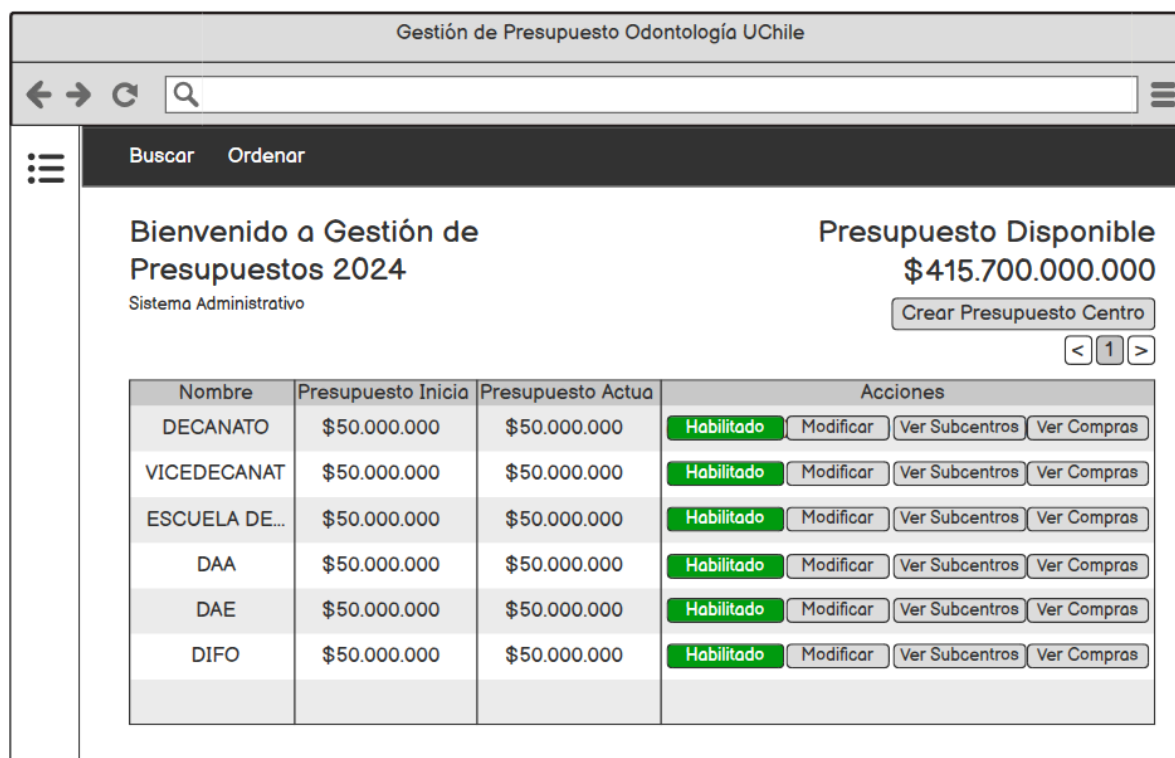


Figura 28. Boceto de Pantalla “Gestión de Presupuesto de Centros de Costo”.

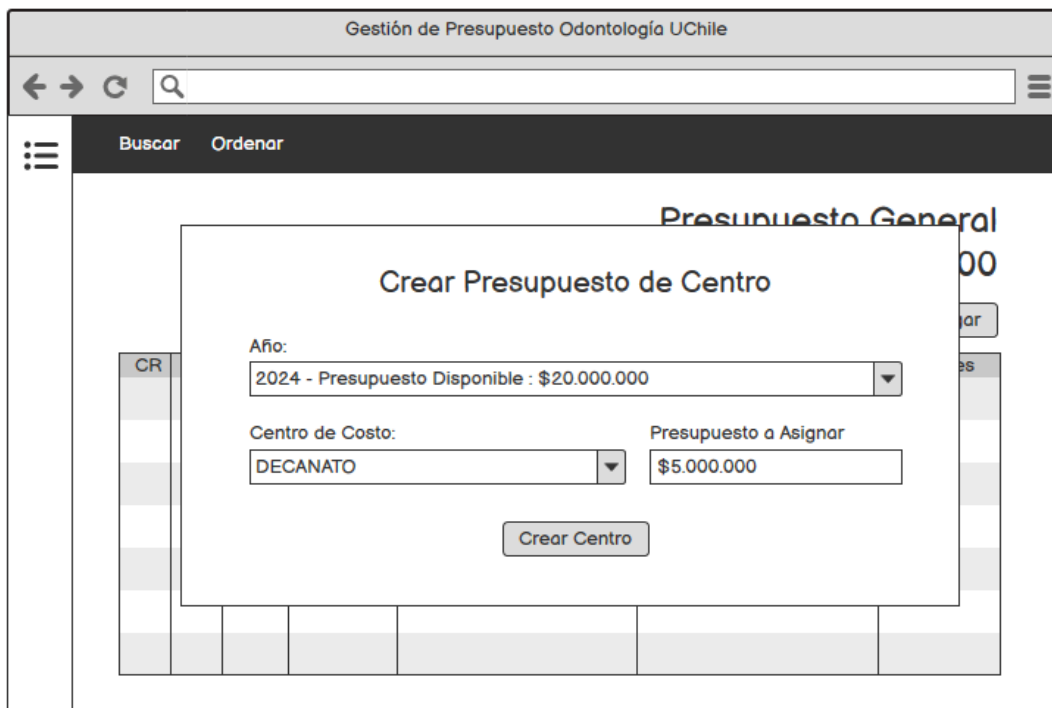


Figura 29. Boceto de Pantalla “Asignación de Presupuesto de Centro de Costo”.

Los usuarios con rol “Responsable” al iniciar sesión, accederán a su Panel de Centro de Costo donde podrán ver la información del presupuesto de su Centro de Costo asignado.

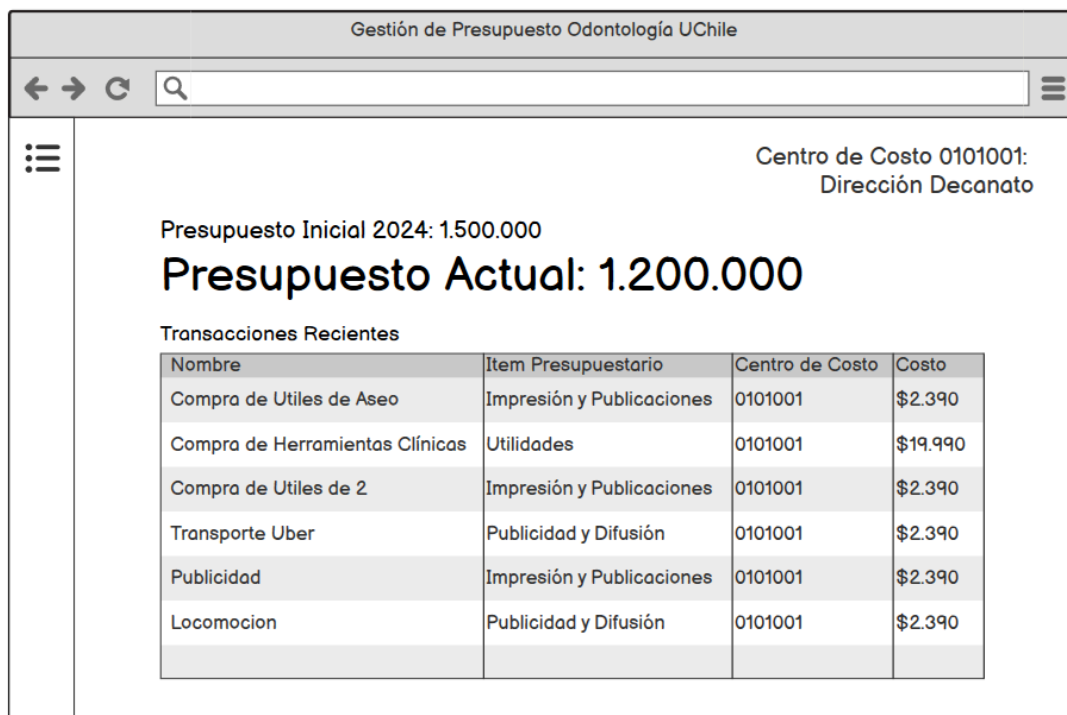


Figura 30. Boceto de Pantalla “Panel de Responsable de Centro de Costo”.

Además los administradores podrán ingresar los detalles de cada compra realizada por la Facultad de Odontología indicando la información del gasto y también el centro de costo que realizó la compra.

Crear Compra

| | |
|---|---|
| N° CDP | Seleccionar Fecha |
| <input style="width: 90%;" type="text" value="N° CDP 4"/> | <input style="width: 90%;" type="text" value="04-01-2024"/> |
| OC Asociada | N° Solicitud |
| <input style="width: 90%;" type="text" value="5610-3-SE24"/> | <input style="width: 90%;" type="text" value="383"/> |
| Nombre de la Adquisición | |
| <input style="width: 95%;" type="text" value="COFFEES PARA EL 5 DE ENERO EN HORARIO AM Y PM. CO"/> | |
| Ingrese Código de Subcentro de Costo | |
| <input style="width: 95%;" type="text" value="1708049"/> | |
| Ingrese Item Presupuestario | |
| <div style="border: 1px solid black; padding: 2px;">510202021 - Movilizacion510202021 - Computacionales510202021 - Artículo Aseo</div> | |

Figura 31. Boceto de Pantalla “Crear Compra”.

6. CONCLUSIÓN

En esta segunda etapa del proyecto de desarrollo del sistema de gestión y control de presupuesto para la Facultad de Odontología de la Universidad de Chile, se ha realizado una correcta modificación con respecto a distintos aspectos como el cambio de la comunicación de los microservicios RabbitMQ por TCP y el aspecto del sistema en base a la retroalimentación recibida tanto por el profesor como por los clientes. La colaboración con el cliente ha sido clave para entender de mejor manera la lógica que se implementará en el sistema contribuyendo en un correcto desarrollo del proyecto.

En términos técnicos, se ha definido la arquitectura del sistema, optando por un enfoque de microservicios utilizando tecnologías como Angular para el frontend y NestJS para los servicios backend. Además de un correcto modelamiento de datos para la construcción de la Base de Datos que utilizará el sistema la cuál permitirá almacenar toda la información necesaria para su correcto funcionamiento.

En resumen, esta segunda etapa del proyecto ha sido de mucha utilidad para ya definir las últimas modificaciones y de esta manera empezar con la implementación. La retroalimentación por parte de los clientes ha sido importante para poder trabajar en conforme a lo que ellos necesiten lo cuál se refleja en un mayor compromiso por parte de ellos y del equipo de trabajo lo que encamina a que el proyecto finalice exitosamente.

7. REFERENCIAS

- [1] Facultad de Odontología. Universidad de Chile. (s. f.). <https://odontologia.uchile.cl/>
- [2] Atlassian. (s. f.). ¿Qué es scrum?. Atlassian. <https://www.atlassian.com/es/agile/scrum>
- [3] Página Web de Bizagi. (2024, febrero). Software gratuito de mapeo y modelamiento de procesos de negocio. Bizagi Modeler. <https://www.bizagi.com/es/plataforma/modeler>
- [4] NestJS - A progressive Node.js framework. (s. f.). NestJS. <https://nestjs.com/>
- [5] MongoDB. (s. f.). MongoDB: the Developer Data platform. <https://www.mongodb.com/>
- [6] Docker: Accelerated Container Application Development. (2024, abril). Docker. <https://www.docker.com/>
- [7] Microservices NestJS. A progressive Node.js framework. (s. f.). <https://docs.nestjs.com/microservices/basics>
- [8] Balsamiq: Fast, focused wireframing for teams and individuals. Balsamiq. (s. f.). <https://balsamiq.com/>
- [9] draw.io. Free flowchart maker and diagrams online. (s. f.). <https://app.diagrams.net/>
- [10] Repositorio de Frontend del Proyecto. (Abril, 2024). <https://github.com/Dylann55/budget-control-system-app-frontend>
- [11] Repositorio de Backend del Proyecto. (Abril, 2024). <https://github.com/Dylann55/budget-control-system-app>