UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA
SAN JOAQUÍN - CHILE

"ANÁLISIS Y DISEÑO DE UNA PLATAFORMA PARA GESTIÓN DE RECURSOS DEL FAB LAB UTFSM."

PATRICIO CARRASCO DONOSO

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO DE EJECUCIÓN INFORMÁTICA

Profesor Guía: Pedro Godoy
Profesor Correferente: Luis Hevia

Octubre - 2019

DEDICATORIA

Mimi, lo hice. Esto es para ti.

AGRADECIMIENTOS

Esta sección sería más larga que la memoria si agradeciera a todas las personas que han tenido una positiva influencia en el desarrollo de este proyecto. Las siguientes personas tuvieron un efecto directo en la victoria que he logrado.

Muchas gracias a las personas que conforman el personal del Fab Lab por permitirme efectuar mi memoria en base a la experiencia empírica que viven en su iniciativa. Particularmente, agradezco a los profesores Diego Aristizabal, quien con paciencia y dedicación me ayudó a superar las dificultades que tuve en mi último ramo de la carrera (Fis120, cof cof), y al profesor Maximiliano Rivera, quien me otorgó las oportunidades.

Muchas gracias a los académicos del Departamento de Informática por los conocimientos brindados a lo largo de mi carrera: Diego Arroyuelo, Hernán Astudillo, Lioubov Dombrovskaia, Claudio Lobos, Jocelyn Simmonds, Marcelo Mendoza y Bernhard Hitpass. Especial agradecimiento a Andrés Moreira y José Luis Martí, quienes se destacaron por su esfuerzo en maximizar el aporte docente a un nivel inspirador.

Muchas gracias a mi profe guía Pedro Godoy, quien aceptó mi idea de memoria, me dio soporte para desarrollarla de manera correcta, y dedicó parte de su tiempo para encaminar y retroalimentar el presente documento. Así que ya saben a quien culpar si tengo algo erróneo.

Muchas gracias a mis amigos, a quienes dejé en el hastío en mis momentos de mayor frustración, pero me soportaron (de milagro): Médano, Hermes, Juanca, Obi-Wan, Karina, Cele, Seba. Después de esto ya puedo retomar los 400++ juegos y Maiden.

Muchas gracias a mi familia, quienes dieron todo y más por asegurarse que tenga éxito: Mi papá y mi mamá, quienes me ofrecieron hasta el Olimpo si me ayudaba a avanzar, la Pau y el Seba, con quienes estudié codo a codo para apoyarnos mutuamente, el Mike quien hacía guardia en mi puerta para asegurarse que estuviera concentrado, mi tío y su familia, quienes me ayudaron en los momentos más difíciles. Sobre todo el Nico, él cantaba el himno de Chile y el Chavo del 8 para ambientar el estudio.

Muchas gracias a ti Cotita: el apoyo incondicional que me diste, noches enteras que me acompañaste mientras avanzaba, el gasto descomunal en comida sana y nutritiva que me entregara los nutrientes necesarios para procesar toda la información contenida en este escrito, fueron mi principal aliento y sustento para darle buen término. Te amo mucho. A ti también Moñoña, cada vez que recibí un mordisco tuyo pude recibir todo el conocimiento necesario.

Muchas gracias a todos uds. por ayudarme de distintas formas a terminar esta larga etapa de forma positiva, ahora puedo dedicar el resto de mi vida a hacer lo que más me gusta, mi vocación: Cultivar papas.

RESUMEN

Resumen— El siguiente documento contiene el análisis y diseño de un sistema web que Fab Lab UTFSM, un espacio de producción de objetos para proyectos de innovación, necesita para cubrir sus necesidades de gestión, en base a los requerimientos de los diferentes stakeholders que se encuentran dentro de sus operaciones diarias. El objetivo es describir todos los procesos, datos y actores involucrados en las actividades actuales, identificar aquellos aspectos que se pueden mejorar y establecer principios, objetivos y otros parámetros directivos que permitan formalizar soluciones que puedan satisfacer los requisitos.

Palabras Clave— Fab Lab, UTFSM, análisis de sistema, diseño de sistema, procesos de negocio, proyecto de desarrollo

ABSTRACT

Abstract— The purpose of this document is to present the analysis and design of a web system that Fab Lab UTFSM, a place to produce objects to facilitate innovative projects, to cover their management needs and build requirements of the different stakeholders inside their daily operations. It records all the data, processes and actors involved in actual activities, it identifies aspects to improve and establish principles, objectives and other parameters to formalize solutions.

Keywords— Fab Lab, UTFSM, system analysis, system design, business processes, development project

GLOSARIO

AGPL (GNU Affero General Public License): Licencia copyleft diseñada específicamente para asegurar la cooperación con la comunidad.
BI (Business Intelligence): Conjunto de estrategias, aplicaciones, datos, productos, tecnologías y arquitectura técnicas enfocados a la administración y creación de conocimiento sobre el medio.
BPMN (Business Process Model and Notation): Notación gráfica estandarizada que permite el modelado de procesos de negocio en formato workflow.
CNC (Computer Numerical Control): Sistema de automatización de machine tools que son operadas mediante comandos programados en un medio de almacenamiento.
CRM (Customer Relationship Manager): Sistemas informáticos de apoyo a la gestión de relaciones con los clientes, venta y marketing.
CRUD (Create, Read, Update and Delete): Refiérase a las funciones básicas en bases de datos o la capa de persistencia de un software.
DEA (Dirección de Ensñanza y Aprendizaje): Departamento dentro de la UTFSM que tiene como misión implementar el modelo educativo y acompañar al profesor en su formación docente.
DIY (Do It Yourself): Cultura de práctica, creación, implementación y reparación de cosas por uno mismo.
DMT (Data Modeling Tool): Software para creación de modelos de datos para sistemas de información.
Fab Lab (Fabrication Laboratory): Referido al espacio de producción de objetos físicos a escala personal o local que agrupa máquinas controladas por ordenadores.
GIG (Grassroots Invention Group): Parte del brazo de investigación de Epistemología y Aprendizaje del Laboratorio de Medios del MIT.
LGPL (GNU Lesser General Public License): Licencia de software creada por la Free Osftware Foundation que pretende garantizar la libertad de comaprtir y modificar el software cubierto por ella, asegurando que el software sea libre para todos sus usuarios.
MIT (Massachusetts Institute of Technology): Universidad privada localizada en Cambridge, Massachusetts.
MVC (Modelo-Vista-Controlador): Patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de su representación y el módulo encargado de gestionar los eventos y las comunicaciones.
RoR (Ruby on Rails): Framework de aplicaciones web de código abierto escrito en Ruby que sigue el paradigma MVC.
RUN (Rol Único Nacional): Número identificatorio único e irrepetible que posee todo chileno y todo extranjero con una visa distinta a la de turista.
UML (Unified Modeling Language): Lenguaje de modelado de sistemas de software.
UTFSM (Universidad Técnica Federico Santa María): Universidad tradicional privada chilena.
XML (Extensive Markup Language): meta-lenguaje que permite definir lenguajes de marcas, utilizado para almacenar datos en forma legible.

ÍNDICE DE CONTENIDOS

RESUMEN iv

ABSTRACT iv

GLOSARIO v

ÍNDICE DE FIGURAS ix

ÍNDICE DE TABLAS ix

INTRODUCCIÓN 1

CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA 2
- 1.1 DESCRIPCIÓN DE LA ORGANIZACIÓN . . . 2
- 1.2 SITUACIÓN ACTUAL . . . 5
  - 1.2.1 ADMISIÓN DE PROYECTOS . . . 5
  - 1.2.2 ADMINISTRACIÓN DE RESERVAS . . . 7
  - 1.2.3 GESTIÓN DE INVENTARIO . . . 9
  - 1.2.4 ASPECTOS A MEJORAR . . . 12
  - 1.2.5 REQUERIMIENTOS DEL USUARIO . . . 12
- 1.3 OBJETIVOS . . . 13
  - 1.3.1 OBJETIVO PRINCIPAL . . . 13
  - 1.3.2 OBJETIVOS ESPECÍFICOS . . . 13
  - 1.3.3 BENEFICIOS . . . 14

CAPÍTULO 2: PROPUESTAS DE SOLUCIÓN 15
- 2.1 ANTECEDENTES PREVIOS . . . 15
- 2.2 ALTERNATIVAS SUGERIDAS . . . 15
  - 2.2.1 PLATAFORMA EXTERNA . . . 16
  - 2.2.2 PLATAFORMA INTERNA MEDIANTE PARAMETRIZACIÓN DE TEMPLATES . . . 18
  - 2.2.3 PLATAFORMA INTERNA A MEDIDA . . . 20
- 2.3 CRITERIOS DE SELECCIÓN . . . 20
- 2.4 METOLODOGÍA DE DESARROLLO . . . 23
- 2.5 GESTIÓN DE RIESGOS . . . 26
- 2.6 HERRAMIENTAS DE SOPORTE PARA EL DESARROLLO . . . 30

CAPÍTULO 3: ANÁLISIS DEL SISTEMA 32
- 3.1 ASPECTOS GENERALES . . . 32
- 3.2 REQUISITOS DEL SISTEMA . . . 33
  - 3.2.1 REQUISITOS FUNCIONALES . . . 33
  - 3.2.2 REQUISITOS NO FUNCIONALES . . . 34
  - 3.2.3 FUERA DE ALCANCE . . . 35

3.3 NUEVOS PROCESOS 35
3.3.1 ADMISIÓN DE PROYECTOS 36
3.3.2 ADMINISTRACIÓN DE RESERVAS 37
3.3.3 GESTIÓN DE INVENTARIO 39
3.4 CASOS DE USO 43
3.4.1 CREAR CUENTA 44
3.4.2 CREAR PROYECTO PROPIO 46
3.4.3 EVALUAR PROYECTO 48
3.4.4 GESTIONAR PROYECTO 50
3.4.5 UNIRSE A PROYECTO EXISTENTE 52
3.4.6 EVALUAR POSTULANTE 53
3.4.7 EFECTUAR RESERVA 56
3.4.8 EVALUAR RESERVA 58
3.4.9 GESTIONAR EQUIPAMIENTO 60
3.4.10 GESTIONAR INSUMO 63
3.4.11 GESTIONAR ESPACIO 66
3.4.12 VER ESTADÍSTICAS E HISTORIAL PARTICIPANTE 69
3.4.13 VER ESTADÍSTICAS E HISTORIAL JEFE DE PROYECTO . 71
3.4.14 VER ESTADÍSTICAS DE ELEMENTOS 73
3.4.15 VER ESTADÍSTICAS FAB LAB 75

CAPÍTULO 4: DISEÑO DEL SISTEMA 78
4.1 MODELO DE DATOS 78
4.2 MODELO FÍSICO 88
4.2.1 ROLUSUARIO 92
4.2.2 USUARIO 93
4.2.3 CARGO 93
4.2.4 USUARIOENPROYECTO 94
4.2.5 ESTADOUSUARIOENPROYECTO 94
4.2.6 TIPOPROYECTO 95
4.2.7 PROYECTO 96
4.2.8 ESTADOPROYECTO 97
4.2.9 RESERVA 97
4.2.10 ESTADORESERVA 98
4.2.11 INSUMOENRESERVA 98
4.2.12 EQUIPAMIENTOENRESERVA 98
4.2.13 UNIDADMEDIDAINSUMO 100
4.2.14 INSUMO 100
4.2.15 TIPOEQUIPAMIENTO 101
4.2.16 ESTADOELEMENTOBODEGUERO 101
4.2.17 EQUIPAMIENTO 102
4.2.18 ESPACIO 103

CAPÍTULO 5: CONCLUSIONES 104

5.1 TRABAJO FUTURO 105

ANEXOS **106**
5.1 NOTACIÓN BPMN 106
5.2 MATRIZ DE TRAZABILIDAD 106
5.3 MODELO LÓGICO 107

REFERENCIAS BIBLIOGRÁFICAS **112**

ÍNDICE DE FIGURAS

1 Logo de Fab Lab UTFSM 2
2 Organigrama esencial del Fab Lab UTFSM. 4
3 Proceso Actual Admisión de Proyectos del Fab Lab UTFSM. 6
4 Proceso Actual Admisión de Reservas del Fab Lab UTFSM. 8
5 Proceso Actual Gestión de Inventario del Fab Lab UTFSM. 10
6 Lista de elementos pertenecientes al Fab Lab. 11
7 Interfaz de Fab. Manager. 16
8 Interfaz de Odoo. 17
9 Interfaz esperada con OpenXava. 19
10 Diagrama de funcionamiento de RoR bajo el esquema MVC. 30
11 Interfaz de plataforma Overleaf, desde su unión con Sharelatex. 31
12 Interfaz de Bizagi Modeler. 31
13 Modelo de Dominio de Plataforma de Manejo de Recursos Fab Lab. . . 33
14 Diagrama de Proceso Admisión de Proyectos con Plataforma Web. . . . 37
15 Diagrama de Proceso Administración de Reservas con Plataforma Web. 38
16 Diagrama de Subproceso de Aprobación de Criterios para Aprobar Reserva. 39
17 Diagrama de Proceso Gestión de Inventario con Plataforma Web. . . . 42
18 Casos de Uso para la plataforma del Fab Lab. 44
19 Diagrama de Secuencia para Crear Cuenta. 45
20 Diagrama de Secuencia en creación de proyectos. 47
21 Diagrama de Secuencia para Evaluar Proyectos. 49
22 Diagrama de Secuencia para Gestionar Proyecto. 51
23 Diagrama de Secuencia para Unirse a Proyecto. 53
24 Diagrama de Secuencia para Evaluar postulación. 55
25 Diagrama de Secuencia para Efectuar Reserva. 57
26 Diagrama de Secuencia para Evaluar Reserva. 59
27 Diagrama de Secuencia para Gestionar Equipamiento. 62
28 Diagrama de Secuencia para Gestionar Insumo. 65
29 Diagrama de Secuencia para Gestionar Espacio. 68
30 Diagrama de Secuencia para ver estadísticas de participante. 70
31 Diagrama de Secuencia para ver estadísticas de Jefe de Proyecto. . . . 72
32 Diagrama de Secuencia para ver estadísticas de Bodeguero. 74
33 Diagrama de Secuencia para ver estadísticas de Operador. 77
34 Modelado ER de los datos a manejar en Fab Lab. 82
35 Modelo general de datos plataforma Fab Lab. 83
36 Diagrama de Clases de plataforma de Fab Lab. 88
37 Modelo Físico de datos plataforma Fab Lab. 92
38 Notación BPMN 2.0 106
39 Matriz de trazabilidad de requerimientos 107
40 Modelo Lógico de datos plataforma Fab Lab. 111

ÍNDICE DE TABLAS

1 Ventajas y desventajas de Fab Lab Manager. 17
2 Ventajas y desventajas de Odoo. 18
3 Ventajas y desventajas de plataforma interna mediante parametrización. 20
4 Ventajas y desventajas de Proyecto Interno In-house. 20
5 Evaluación de aspectos beneficiosos para plataforma externa. 21
6 Evaluación de aspectos riesgosos para plataforma externa. 21
7 Evaluación de aspectos beneficiosos para plataforma externa mediante templates. 22
8 Evaluación de aspectos riesgosos para plataforma externa mediante templates. 22
9 Evaluación de aspectos beneficiosos para plataforma externa hecha a medida. 22
10 Evaluación de aspectos riesgosos para plataforma externa hecha a medida. 22
11 Promedio de aspectos positivos y negativos evaluados para cada solución. 23
12 Costos de la solución desarrollada en RoR. 24
13 Planificación de desarrollo para plataforma. 25
14 Riesgo Técnico 1 26
15 Riesgo Técnico 2 27
16 Riesgo de Proyecto 1 27
17 Riesgo de Proyecto 2 28
18 Riesgo de Proyecto 3 28
19 Riesgo Personal 1 29
20 Riesgo Personal 2 29
21 Tabla de Rol de Usuario 92
22 Tabla de Usuario 93
23 Tabla de Cargo en Proyecto 93
24 Tabla de Usuario en Proyecto 94
25 Tabla de Estados de Usuario en Proyecto 94
26 Tabla de Tipos de Proyecto 95
27 Tabla de Proyecto 96
28 Tabla de Estados del Proyecto 97
29 Tabla de Reserva 97
30 Tabla de Estados de la Reserva 98
31 Tabla de insumos en reserva 98
32 Tabla de equipamiento en reserva 99
33 Tabla de unidades de medida de insumo 100
34 Tabla de insumo 100
35 Tabla de Tipos de Equipamiento 101
36 Tabla de Estados de Elemento de Bodeguero 101
37 Tabla de Inventario 102
38 Tabla de Espacio 103

INTRODUCCIÓN

Un Fab Lab es un espacio o taller de producción de objetos físicos a pequeña escala, que tiene como objetivo la elaboración de "cualquier cosa"[Gershenfeld, 2012]. El concepto principal detrás de esta iniciativa es el aprendizaje práctico o "aprender haciendo", dando los recursos a estudiantes para resolver problemas y necesidades particulares y crear aparatos inteligentes propios. Nace de la colaboración conjunta entre el grupo de investigación de Epistemología y Aprendizaje del Laboratorio de Medios, y el Center for Bits and Atoms, proyectos interdisciplinarios del MIT que explora los límites entre ciencia computacional y física.

Los Fab Labs se equipan con máquinas y herramientas controladas por ordenadores que permiten tratamiento de materiales de diferente naturaleza y tamaño, como por ejemplo máquinas de fresado, de corte con láser, de control decimal numérico, entre otros aparatos que permiten prototipado ágil.

El presente trabajo se centra en el Fab Lab de la Universidad Técnica Federico Santa María, una idea en desarrollo que se enfoca en proyectos multidisciplinarios con énfasis en la innovación y colaboración entre estudiantes de diferentes carreras. En la búsqueda de cumplir con todos los requerimientos para ser reconocidos como un Fab Lab oficial, se han visto enfrentados a problemas de índole práctico que tienen directa relación con la gestión de los recursos en el ofrecimiento de sus servicios, provocando principalmente una falta de información relevante de lo que poseen, una ejecución ineficiente de sus tareas y dificultad para mejorar los procesos. Se busca integrar y gestionar la información más relevante de una manera óptima, así como mejorar las labores de los diferentes encargados por medio de una plataforma web.

Este documento contiene el estudio llevado a cabo de la organización, incluyendo una revisión a su estructura organizacional, objetivos, servicios ofrecidos, y cómo se ejecutan sus procesos principales. Se proponen soluciones, herramientas y procesos nuevos que se adapten a la organización junto con un amplio análisis a sus necesidades.

CAPÍTULO 1
DEFINICIÓN DEL PROBLEMA

Este capítulo abarca los datos básicos de la empresa a estudiar y analizar los aspectos que necesitan mejora dentro de su situación actual. Se presenta una breve descripción de la organización, el contexto bajo el cual realizan sus actividades diarias, los potenciales puntos que se pueden mejorar y los beneficios de efectuar medidas, para así establecer objetivos claros a cumplir con la solución.

1.1. DESCRIPCIÓN DE LA ORGANIZACIÓN

Fab Lab UTFSM, cuyo logo se muestra en la figura 1, se define a si mismo como un laboratorio de prototipado y prueba de conceptos[UTFSM, 2018] que nace de la necesidad de brindar un espacio de desarrollo de proyectos multidisciplinarios en el Campus San Joaquín de la Universidad Técnica Federico Santa María, ubicado en la Avda. Vicuña Mackenna, al constatar la falta de formulación y desarrollo de iniciativas que involucran cooperación de diferentes carreras[FabLab, 2018]. Su principal objetivo es brindar el conocimiento y los recursos a cualquiera que busque aprender, innovar e inventar utilizando la fabricación digital fomentando proyectos que apunten a la innovación, educación, tecnología y colaboración con una base científica[Rivera, 2018]. Buscan ser a futuro un servicio a nivel regional que permita a la comunidad de estudiantes, académicos, investigadores y personas interesadas en fabricación de diseños y propuestas que utilicen sus espacios.

Figura 1: Logo de Fab Lab UTFSM
Fuente: Fab Lab UTFSM Website.

Para cumplir con el objetivo principal, Fab Lab UTFSM realiza los siguientes servicios:

- **Reservas de bienes y espacio a proyectos**: El propósito principal es otorgar a estudiantes derecho al uso de los recursos, espacio y personal que posee el Fab

Lab para la realización de tareas dentro del marco de los proyectos que deseen desarrollar.

- **Talleres de capacitación**: La realización de talleres de carácter educativo dentro de los espacios del Fab Lab tienen por objetivo introducir de manera práctica la maquinaria disponible, tal que permita un acercamiento de los estudiantes y genere interés en desarrollar proyectos propios.

Fab Lab UTFSM dispone de espacios dentro de la universidad, así como de diversos equipos tales como máquinas CNC, impresoras 3D de última generación, impresora de circuitos, cortadora laser, junto con materiales, herramientas básicas y elementos electrónicos que se distribuyen entre los proyectos[UTFSM, 2017]. Se dan a conocer a través de redes sociales tales como FaceBook[1], donde suelen organizar sus talleres, eventos, e información importante a divulgar, Instagram[2], donde publican tanto imágenes y videos del funcionamiento interno del Fab Lab y sus máquinas, y en redes de Fab Lab internacionales[3].

Son la cuna de diferentes iniciativas exitosas tales como:

- **Guante Khapto**: Dispositivo de mediciones enfocado en diagósticos y evaluaciones de kinesiólogos.
- **PrevUPP**: Dispositivo que previene las úlceras por presión en discapacitados.

También son una fuente de acercamiento hacia diferentes herramientas de prototipado, aplicaciones y forma de utilización de una gran variedad de herramientas presentes mediante talleres y eventos dentro de la universidad.

El esquema presente en la figura 2 presenta los roles que ocupan las distintas personas que conforman el equipo actual del Fab Lab, los cuales son los siguientes:

- **Lab Director**: Es el representante visible del Fab Lab ante las autoridades de la universidad y posibles inversionistas. Actúa como mediador al momento de rendir cuentas y solicitar permisos específicos.
- **Lab Manager**: Sus responsabilidades están relacionadas con la supervisión de los proyectos y sus avances. Establece y verifica condiciones comunes bajo las cuales el grupo que desarrolla un proyecto puede solicitar y usar los recursos del Fab Lab.

[1] `https://www.facebook.com/fablabusm/`
[2] `https://www.instagram.com/fablab_utfsm/`
[3] `https://www.fablabs.io/labs/fablabutfsm`

- **Operations Administrator**: Se encarga de llevar cuentas de índole económica dentro del Fab Lab, registrando gastos e ingresos de las actividades que se efectúen. También gestiona máquinas y componentes, servicios computacionales y de red.
- **Development Administrator**: Verifica las necesidades de los proyectos, recomienda procedimientos y herramientas que permitan su desarrollo de manera eficiente. También es el vínculo con los alumnos al Fab Lab en ámbito comunicacional, otorgando informacón del funcionamiento de éste.
- **Support Staff**: Encargados de apoyar las funciones ejercidas por el Lab Manager, Operations y Development Administrators.

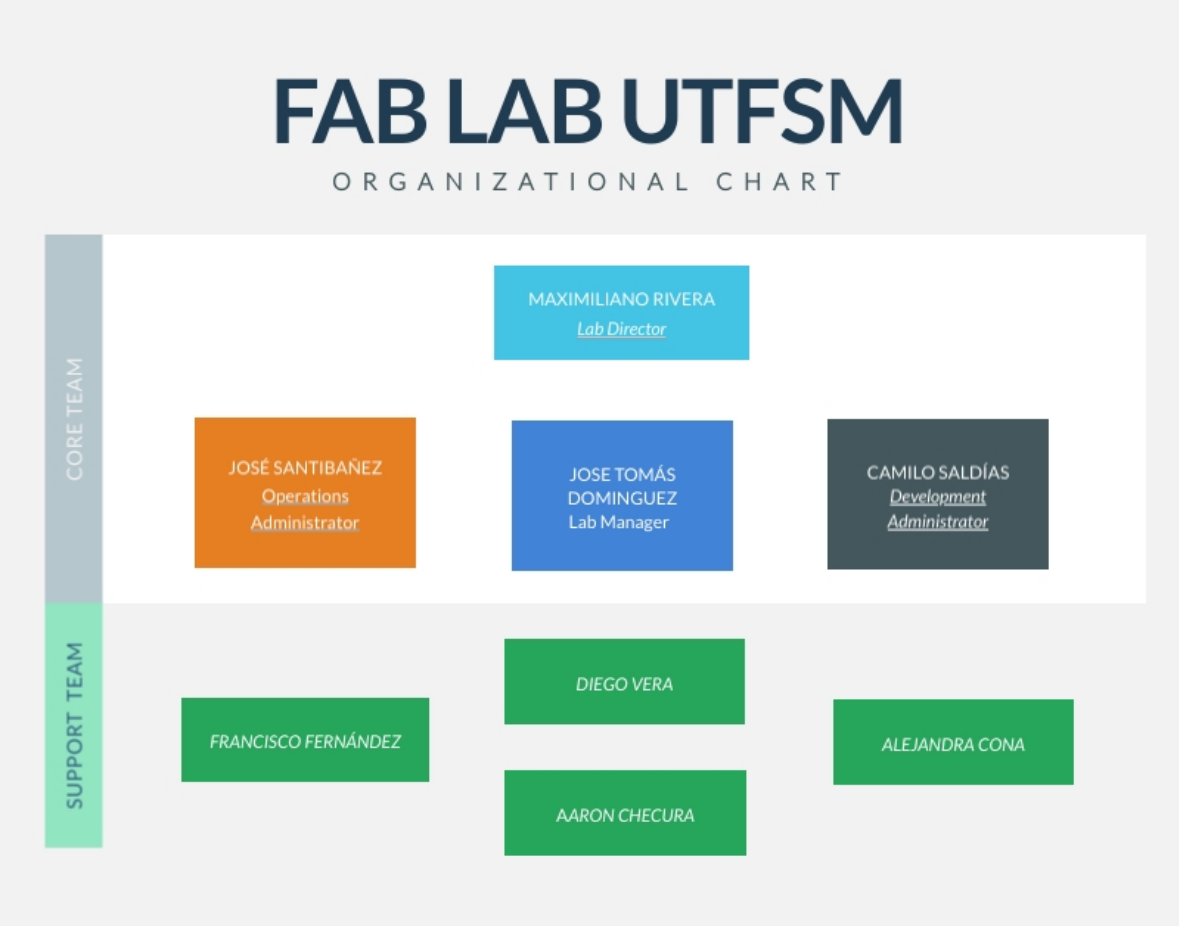


Figura 2: Organigrama esencial del Fab Lab UTFSM.
Fuente: Elaboración Propia

Notar que estos roles son de carácter referencial, dado que las funciones que desempeñan cada uno puede variar según aparezcan nuevos quehaceres.

1.2. SITUACIÓN ACTUAL

Los servicios del Fab Lab están pensados de manera inicial para estudiantes, académicos e investigadores que se desenvuelven dentro de la comunidad de la UTFSM que estén interesados en llevar a cabo proyectos de innovación, donde los fines y condiciones de estos proyectos son definidos a consenso mediante acuerdos y convenios previos. El funcionamiento y gestión del Fab Lab se encuentra en constante construcción y mejora, ya que se organizan e implementan actividades o datos a medida que se necesitan o que surgan riesgos y obstáculos no planificados.

Los procesos principales que se analizan son los siguientes:

- **Admisión de Proyectos**: Todas las personas que consideren crear y llevar a cabo proyectos con recursos del Fab Lab deben pasar por etapas que involucran inscripción y revisión de la idea que quieren llevar a cabo.
- **Administración de Reservas**: El personal del Fab Lab otorga derecho a solicitar y utilizar máquinas, dispositivos, herramientas eléctricas, insumos, componentes y otros elementos a disposición según ciertos criterios básicos a los proyectos que llegan. Permiten incluso préstamos por períodos mayores fuera de sus espacios.
- **Gestión de Inventario**: La adquisición, revisión, actualización de elementos pertenecientes al Inventario del Fab Lab, que tienen por objetivo servir al desarrrollo de los proyectos, son manejados por el personal del Fab Lab de tal manera de asegurar disponibilidad para reservas.

1.2.1. ADMISIÓN DE PROYECTOS

Para que los participantes presenten sus propuestas de proyectos, el personal del Fab Lab entrega a disposición un formulario en Google Forms, cuyo acceso se otorga mediante código QR en el frontis, directamente como link en el interior de las instalaciones o por redes sociales. Este formulario se divide en dos secciones: La primera sección consulta datos generales como el nombre, resumen ejecutivo, datos del encargado, integrantes y tipo de proyecto, cuyas opciones son: Académico, Iniciativa Estudiantil, Proyecto para Empresa Particular y Proyecto Personal.

El tipo de proyecto determina la segunda sección del formulario, ya que se efectúan preguntas con énfasis personalizado:

- **Académico**: Profesor a Cargo, Departamento y nombre de asignatura.
- **Iniciativa Estudiantil**: Departamento, persona a cargo, mencionar si se han adjudicado algún fondo, así como mencionar disposición a publicar proyecto bajo licencia Open Source u Open Hardware.

- **Empresa Particular**: Tipo de Empresa, años de existencia, detallar si empresa está legalmente constituída y cuál es el vínculo con la universidad.
- **Proyecto Personal**: Detallar objetivos, financiamiento y mencionar disposición a publicar proyecto bajo licencia Open Source u Open Hardware.

Cuando el personal del Fab Lab revisa los datos ingresados por el interesado, notifica presencialmente a los miembros del proyecto la aprobación de su idea. Actualmente no se manejan criterios de aceptación fijos o requerimientos mínimos de aprobación para ejecutar proyectos en el Fab Lab, ya que buscan aumentar el flujo de proyectos, y se negocia de manera personalizada cómo el Fab Lab se beneficia del proyecto. Esto puede ser: Efectuar voluntariados, distribución libre de conocimiento generado por el proyecto, marketing de boca a boca del Fab Lab, entre otros. Por lo que se concluye que todo proyecto resulta eventualmente aprobado, permitiendo a sus miembros tener el derecho a reservar los bienes y/o espacios que poseen.

El proceso se resume como muestra la figura 3

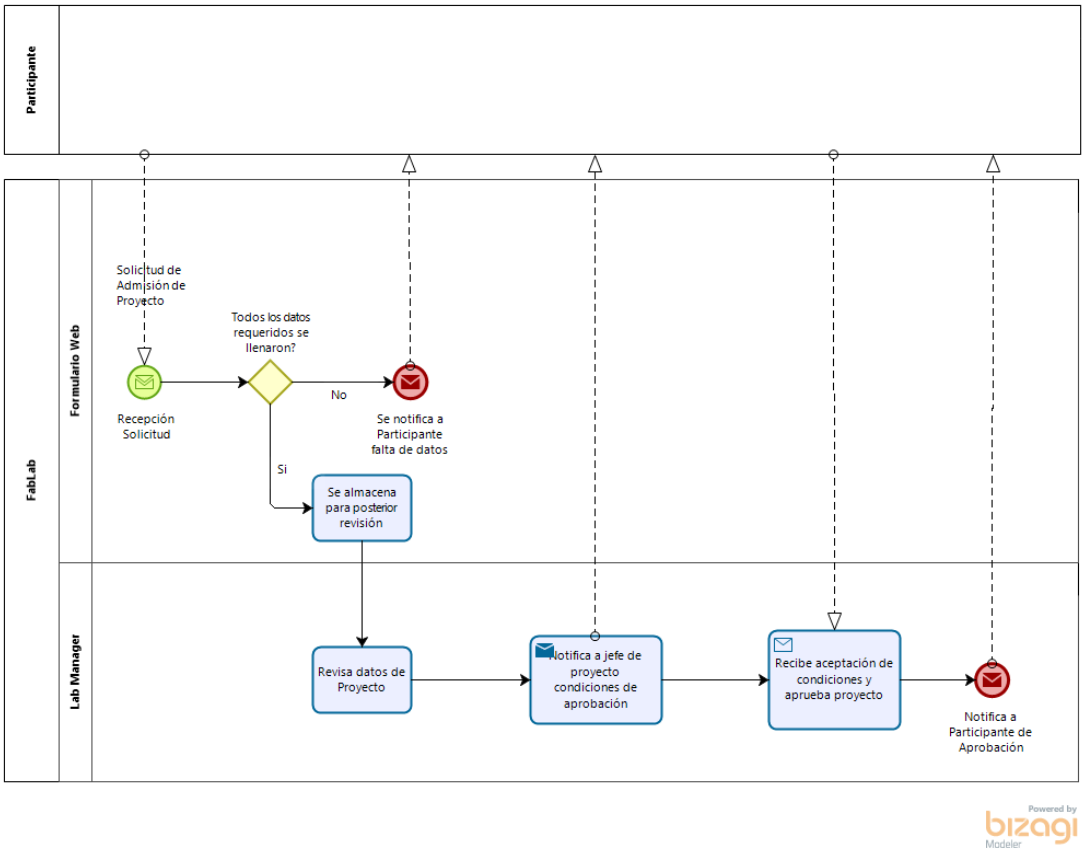


Figura 3: Proceso Actual Admisión de Proyectos del Fab Lab UTFSM.
Fuente: Elaboración Propia

1.2.2. ADMINISTRACIÓN DE RESERVAS

Una vez los proyectos son aceptados, los integrantes tienen derecho a efectuar reservas. Cualquier integrante se presenta en los espacios del Fab Lab, donde solicitan el equipamiento, insumo y espacios necesarios para sus tareas. El personal los atiende y verifica los siguientes requisitos:

- **Nivel de permiso**: El equipamiento tiene niveles de permiso según el ítem a reservar, donde las máquinas tienen el nivel más alto, requiriendo pasar por un taller de capacitación antes de poder usarlas. Si el equipo a reservar presenta esta limitación, se le consulta al solicitante si efectivamente asistió a uno. Si la respuesta es negativa, se solicita que asista a un taller si requiere efectuar esta reserva.
- **Disponibilidad**: El personal verifica físicamente que existan elementos disponibles que se desean reservar, lo que implica que no estén en uso ni en mantención. Si esta verificación resulta con elementos que cumplan con este requisito, la reserva se aprueba. En caso negativo, se ofrece notificar cuando se encuentren disponibles, registrando al equipo en una lista de espera presente en un cuaderno. Sin embargo, este medio es usado de manera ocasional.

Fab Lab no posee un mecanismo de control estandarizado que permita dar a conocer qué se reserva, a quién se reserva, quién es la persona que autoriza esa reserva, tiempo en que se lleva a cabo ni estado de ésta en algún período determinado. De manera esporádica se registran las reservas y algunos de estos datos en formularios físicos o anotaciones en pizarras.

El diagrama presente en la figura 4 es un compendio de la secuencia de tareas llevadas a cabo cuando se solicita una reserva.

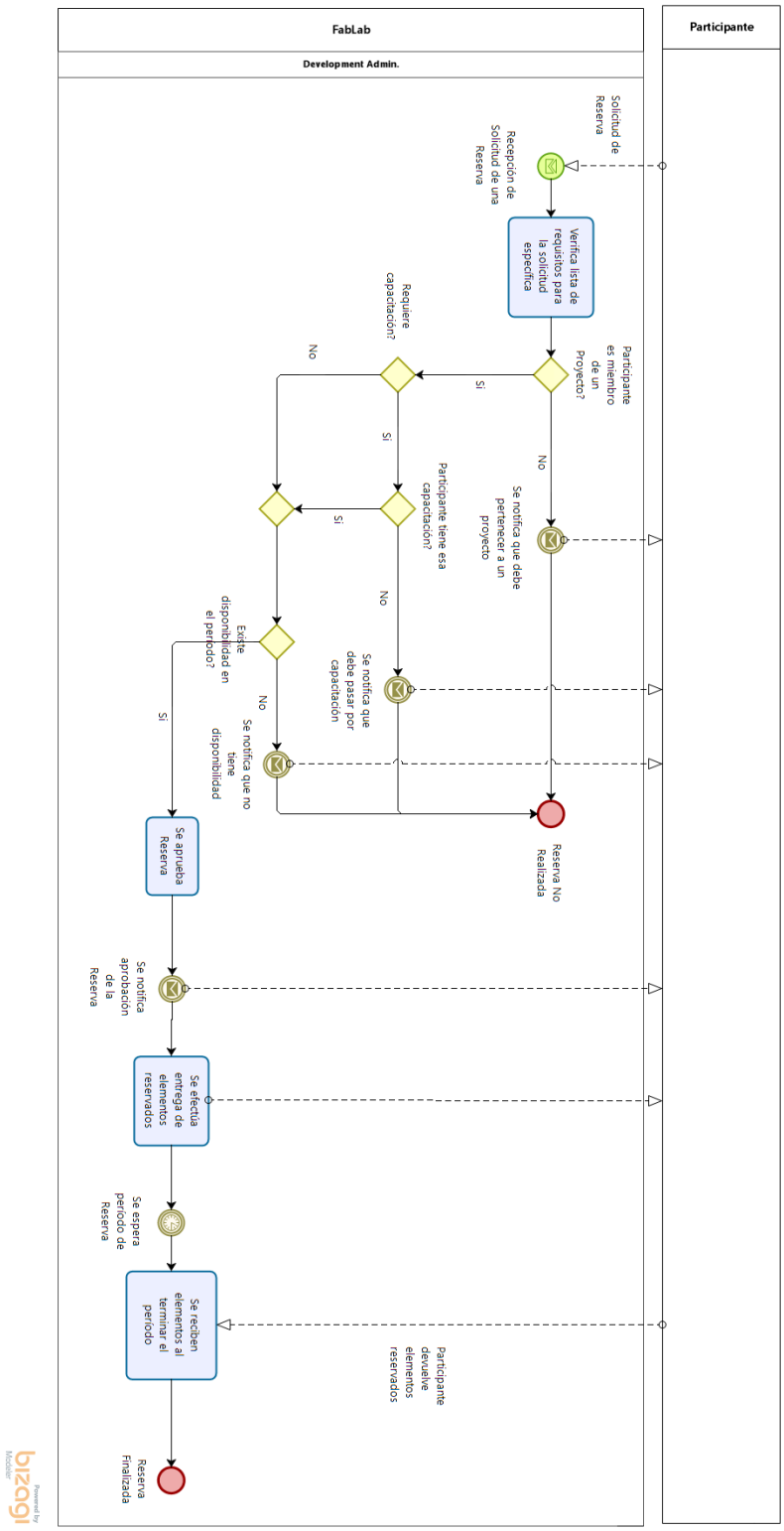


Figura 4: Proceso Actual Admisión de Reservas del Fab Lab UTFSM.
Fuente: Elaboración Propia

1.2.3. GESTIÓN DE INVENTARIO

Existen tres subprocesos principales que se efectúan en paralelo de manera periódica: la compra y adquisición de elementos para inventario, la mantención de los ítems dañados y la entrega de elementos para reservas aprobadas por staff del Fab Lab mediante el proceso descrito en la sección 1.2.2. La manera en que se organizan estos subprocesos paralelos se presentan en la figura 5.

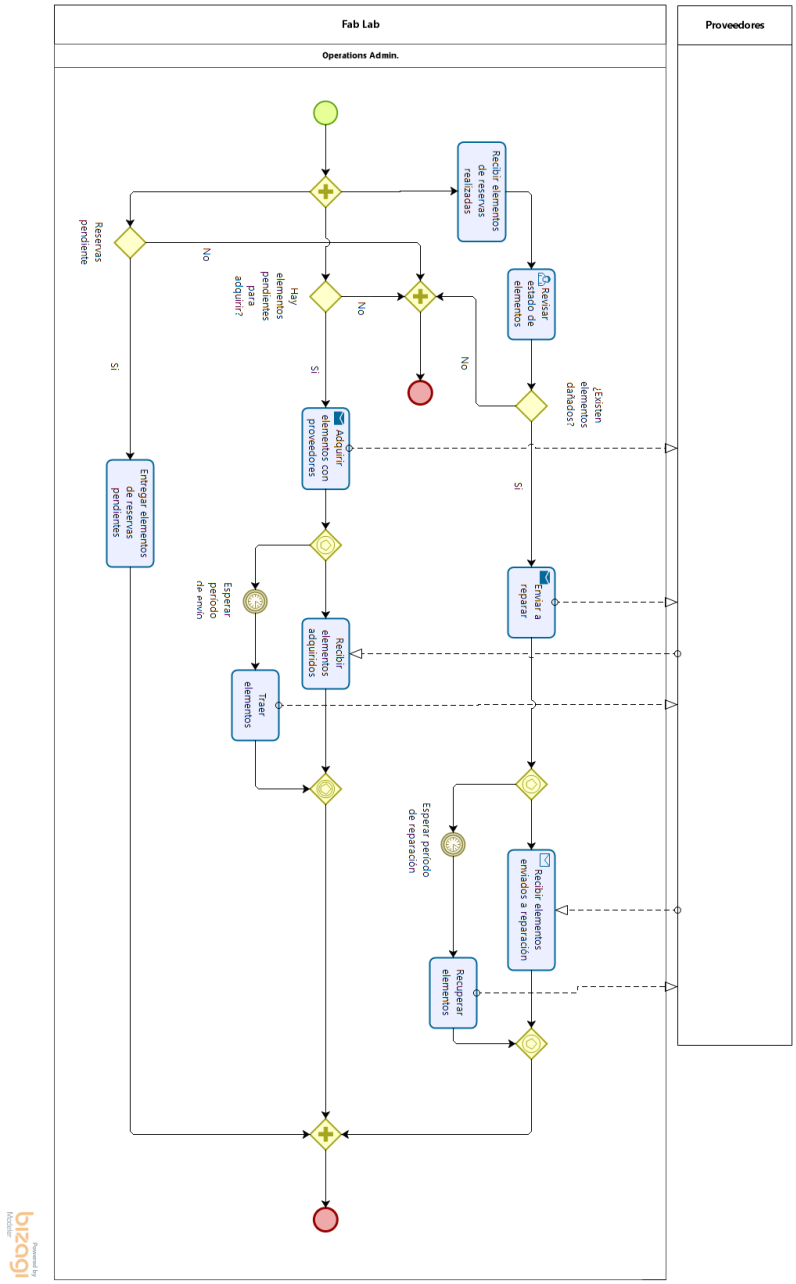


Figura 5: Proceso Actual Gestión de Inventario del Fab Lab UTFSM.
Fuente: Elaboración Propia

Todos los elementos que actualmente posee el Fab Lab, sean máquinas, componentes, materiales, piezas, entre otros, se encuentran registradas en una hoja de cálculo online en Google Drive, la cual se actualiza conforme van adquiriendo productos, o los existentes sufren cambios.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Cant	Tipo	Descripción	Nº Serie	Cód Inventario USM	Dueño?	Prestado a:	Observaciones						
168	6	Materiales	Conectores atornillables para Breadboard Verde											
169	10	Materiales	LED color Verde											
170	17	Materiales	LED color Rojo											
171	1	Materiales	Motor con Caja Reductora 6V 105RPM					http://www.dx.com/p/chihaimotor-12mm-12gagb-dc6-0v-105rpm-precision-deceleration-gear-motor-silver-339630?tc=CLP/						
172	1	Materiales	Teclera c/Botones numéricos + A-B-C-D-*-#											
173	1	Materiales	LED color Amarillo											
174	1	Herramientas	Sierra manual c/hoja											
175	1	Materiales	Alcohol desnaturalizado 70% 125ml											
176	1	Materiales	Laca para Impresora 3D V05 412ml											
177	1	Materiales	Placa para soldar CL-008 36x55											
178	1	Materiales	Hilo conductor plateado											
179	1	Motores	Servo Hitec HS81											
180	1	Materiales	Soportes Pan/Tilt para Servo					Incompleto? https://www.olimex.cl/shop/product/soportes-pan-tilt-10275?search=Automatización						
181	1	Materiales	Display 20x2 Winstar WH2002A-YGH-ET					http://www.winstar.com.tw/products/character-lcd-display-module/lcd-20x2.html						
182	1	Materiales	Carrete de hilo de Nylon transparente											
183	14	Materiales	Potenciómetro para Breadboard 10KOhm (P 103 122 C)											
184	1	Materiales	Pasta para soldar Indepp											
185	1	Materiales	Pasta para soldar Pro's Kit 8S005											
186	1	Fuente de Poder	Transformador 14.5V 530mA 6971412 FW4798/A											
187	1	Materiales	Módulo SRAM y Timekeeper M48T35AV-10PC1					http://www.mouser.com/ds/2/389/m48t35av-955079.pdf						
188	1	Materiales	Memoria EPROM 512KB M27C512					http://www.mcropik.com/PDF/27C512.pdf						
189	1	Materiales	7-ch Darlington Sink Driver ULN2004APG					http://owidu.geekallens.com/wp-content/uploads/sites/2/2016/06/ULN2003APG.pdf						
190	1	Materiales	Transformador para 10/100 Ethernet H161010DFG					http://www.mag-tek.com.tw/downloadbox/100single-m/H1610XDF0.pdf						
191	1	Materiales	Amplificador de audio de bajo voltaje LM386N					http://www.ti.com/lit/ds/symlink/lm386.pdf						
192	10	Materiales	Disipadores de Aluminio negro											
193	1	Materiales	Disipador de Aluminio Verde											
194	2	Materiales	Parlante piezoeléctrico											
195	4	Materiales	Power MOSFET IRF744N					http://www.vishay.com/docs/91056/91056.pdf						
196	1	Materiales	Rectificador de propósito general D15XB60					http://pdf.datasheetcatalog.com/datasheet/Shindengen/mXxazxs.pdf						
197	2	Materiales	Tiristor SK M1261											
198	1	Materiales	Rectificador de alta velocidad C93M-03											
199	1	Materiales	Tiristor SJ F861											
200	1	Materiales	Rectificador C83M-006											
201	2	Materiales	Regulador de tensión lineal L7805CV											

Figura 6: Lista de elementos pertenecientes al Fab Lab.
Fuente: Fab Lab UTFSM, Google Hojas de Cálculo.

Entre los datos que registran se encuentran:

- **Cantidad**: Número de veces que tienen cierto material.
- **Tipo**: Categoría asignada al elemento. Se identifican, por ejemplo, materiales, motores, piezas mecánicas, documentación, entre otros.
- **Descripción**: Se explica de manera general el elemento.
- **Número de Serie**: Código alfanumérico con el que identifica un único elemento el Fab Lab de manera interna.
- **Código de inventario USM**: Número que asigna la universidad a un elemento que el Fab Lab recibe.
- **Dueño**: Registra si el elemento en cuestión pertenece a un particular en lugar del Fab Lab.
- **Prestado a**: Si el elemento lo tiene alguien, este campo tiene su nombre completo.
- **Observaciones**: Notas que especifican propiedades del elemento. Puede variar, siendo posible poner un enlace, estado del elemento o características específicas.

Algunos de estos datos no son necesariamente registrados, como por ejemplo código de inventario USM, ya que la universidad no etiqueta todos los elementos que llegan al inventario del Fab Lab. Existen pocos registros que tengan atributo 'dueño' y el atributo 'Prestado a' se detalla en múltiples formatos.

1.2.4. ASPECTOS A MEJORAR

Las necesidades que se han creado en el Fab Lab a medida que ha aumentado el ingreso de proyectos tienen que ver con la mejora y consolidación de las bases de organización general, que permita proveer sus servicios de manera más rápida, efectiva y estructurada, y en paralelo, sacar provecho del uso que los proyectos le dan al Fab Lab para estudiar en manera más profunda de qué manera se utilizan los recursos y asignar métricas que los midan. En los procesos actuales se han identificado los siguientes problemas:

- **Anomalías de información**: Efectuar reservas con formularios físicos para la gestión de préstamos y reservas causa múltiples problemas; como el proceso no está estandarizado, miembros del personal del Fab Lab pueden digitalizar información repetida, causando redundancia. También se presenta información inconsistente [Martí, 2012], que impide tener un seguimiento de los elementos reservados en el Fab Lab, las personas que reservaron, por cuánto tiempo y otros datos de relevancia que son básicos para identificar una reserva. Como estos aspectos dificultan cualquier estudio analítico del uso de recursos del Fab Lab, tiene como consecuencia no poder transparentar a stakeholders el impacto real que tienen sus servicios, ni tampoco pueden estudiar el nivel de utilización.
- **Escalabilidad**: A medida que exista más personas interesadas en utilizar los servicios del Fab Lab, el registro y almacenamiento de la información es más engorroso conservando la metodología de formularios físicos. La falta de centralización e integración[Veltman, 2013] de los datos no permite adaptarse a posibles mejoras, nuevos servicios y funciones sin sacrificar tiempo adicional en ordenar, clasificar y almacenar nuevos datos. Esto se vuelve más complicado cuando es necesario modificar datos que ya existen, por ejemplo, una reserva a futuro que quiere hacerse en período distinto al inicial. Si no se registra, no se puede monitorear, y si se efectúa en formularios físicos, se complican los cambios.
- **Fallas de seguridad**: Dada la poca formalidad en asegurar ciertos requisitos implícitos en la formación de los miembros de proyectos al utilizar los servicios del Fab Lab, existe poca delimitación en el acceso y uso de la infraestructura y equipo, lo que ocasiona poca delimitación de quienes pueden usar el Fab Lab y confusas condiciones bajo las cuales se ocupan los recursos.

1.2.5. REQUERIMIENTOS DEL USUARIO

Las funcionalidades buscadas por el Fab Lab se encuentran priorizadas de acuerdo a aspectos funcionales de corto plazo que sean de influencia directa en las operaciones cotidianas que se llevan a cabo, modularizando las características y opciones buscadas en la plataforma. A continuación se presenta un resumen de los requisitos pedidos por el usuario en base a las entrevistas realizadas a los stakeholders respectivos.

- Personas que sean miembros de un proyecto tienen la capacidad de efectuar reservas de los recursos que tiene el Fab Lab.
- Los usuarios nuevos que se registren debiesen ser capaz de crear sus proyectos en la plataforma y postular a proyectos creados por otras personas.
- Debe permitir a las personas encargadas de sus proyectos administrar los datos que registren de ellos.
- El Staff del Fab Lab debe tener acceso al registro generado por las reservas efectuadas, y visualizar estadísticas del uso del Fab Lab.
- Debe permitir a las personas encargadas del Fab Lab administrar el inventario, maquinaria y espacios que tienen disponibles.
- Se busca que las funcionalidades descritas puedan ser ejecutadas desde cualquier dispositivo de uso común que tenga conexión a internet. Esto puede ser un computador, smartphones o tablets, por ejemplo.

1.3. OBJETIVOS

Se establecen los siguientes objetivos a lograr con el desarrollo del proyecto.

1.3.1. OBJETIVO PRINCIPAL

Se busca como objetivo principal analizar y diseñar una aplicación que sustituya de manera gradual las plataformas que operan actualmente para llevar a cabo admisión de proyectos, administración y gestión de uso de equipamiento e insumos y reservas, logrando los siguientes valores añadidos:

- Solucione los problemas derivados de las anomalías de información mencionadas en la sección 1.2.4.
- Mejora de la escalabilidad de la plataforma y seguridad del uso del Fab Lab.

1.3.2. OBJETIVOS ESPECÍFICOS

Los Objetivos específicos son los siguientes:

- Mejorar la captación, almacenamiento y análisis de datos recolectados por el Fab Lab, reduciendo la tasa de datos perdidos o incompletos.

- Establecer tareas, canales y roles mejor estandarizados dentro de los procesos, tal que disminuya los tiempos de ejecución.
- Generar métricas que den cuenta del uso de Fab Lab.

1.3.3. BENEFICIOS

Se esperan como mayores beneficios esperados de la realización de este proyecto los siguientes:

- Ejecución de tareas en menor tiempo.
- Potencial analítico del uso del Fab Lab.
- Estructuración y estandarización de procesos esenciales y datos generados.
- Mayor centralización de datos.
- Mayor transparencia de resultados.
- Integración relacional y consistente de los datos.
- Mayor control y monitoreo de acceso a los recursos.

CAPÍTULO 2
PROPUESTAS DE SOLUCIÓN

Esta sección describe de manera general las posibles soluciones a implementar para solucionar los problemas descritos, las restricciones a las que se somete cada una de ellas y los criterios de selección para definir cual es la propuesta a utilizar.

2.1. ANTECEDENTES PREVIOS

Dados los aspectos identificados que se pueden optimizar, se establece la necesidad de desarrollar una plataforma web que maneje esta lista de mejoras en el marco de un sistema de información, tal que reemplace e integre las herramientas actuales y conlleve una actualización de sus procesos. Si bien la base de este proyecto debe considerar desarrollo de una serie de funcionalidades de manera incremental, se otorga prioridad a los procesos de índole práctica. Esta sección presenta las posibles alternativas de solución y los criterios de selección para escogerla.

Para plantear estas alternativas, se consideran las siguientes restricciones:

- La plataforma debe ser de bajo costo, preferentemente uso de herramientas open source: Condicionar el desarrollo de esta manera apoya el uso de software open source, genera código que puede ser compartido, reduce costos monetarios por licencia y fomenta los valores bajo los cuales trabaja el Fab Lab.
- Se exige solución con enfoque modular y que facilite la integración de futuros colaboradores: Esta restricción tiene en consideración que las funcionalidades a desarrollar en esta fase del proyecto son un punto inicial a mejorar gradualmente en el largo plazo, y a la vez, tiene el potencial de ser una iniciativa impulsada por los mismos miembros que componen el Fab Lab.

Adicionalmente, dado los objetivos expuestos en la sección anterior, esta aplicación debe ser accesible por un navegador web.

2.2. ALTERNATIVAS SUGERIDAS

Las posibles soluciones que se pueden utilizar para satisfacer los objetivos vistos en la sección 1.3 son las siguientes:

2.2.1. PLATAFORMA EXTERNA

Se toma una solución ya desarrollada de manera externa por personas que hayan tenido problemas similares, y se implementa localmente. Entre las soluciones se tiene:

- **Fab. Manager**[4]: Es una de las únicas soluciones personalizadas para un fablab desarrolladas con Ruby on Rails y AngularJS. Está disponible bajo la licencia AGPL cuyo código se encuentra visible en GitHub y posee una demo disponible. Tiene funcionalidades adaptadas según usuario como se puede apreciar en la figura 7. El cliente puede acceder a funciones tales como:
  - Reservar una máquina.
  - Inscripción de talleres.
  - Reservar un espacio.
  - Galería de proyectos.

Figura 7: Interfaz de Fab. Manager.
Fuente: `https://demo.fab-manager.com/`

Ocupando las credenciales de "admin", algunas de las opciones existentes son las siguientes:

- Monitorear talleres.
- Administrar Usuarios.
- Manejar eventos.

[4] `https://www.fab-manager.com/?lang=en`

- Organizar Espacios.
- Gestión de máquinas.

Esta solución no efectúa funcionalidad relacionada con la extracción, análisis de datos de uso del equipamiento y espacio, ni reportabilidad de los aspectos anteriores perteneciente al Fab Lab.

En resumen, las ventajas y desventajas de esta solución se encuentran en la tabla 1:

Ventajas	Desventajas
Varios requerimientos ya desarrollados	Análisis de datos no presente
Menor esfuerzo de desarrollo	Soporte dependiente de la comunidad nicho
Solución probada en otros Fab Labs	Poca adaptabilidad a necesidades locales

Tabla 1: Ventajas y desventajas de Fab Lab Manager.
Fuente: Elaboración Propia

- **Odoo**[5]: Software empresarial que incluye CRM, comercio electrónico, facturación, contabilidad, gestión de almacenamiento y gestión de proyectos. Este software posee 2 versiones: Odoo Community, versión que incluye código abierto, y la segunda es Odoo Enterprise la cual complementa la primera versión ya mencionada con características y servicios comerciales. Posee una demo web donde se podrá probar de forma inmediata sus aplicaciones principales. Su versión open source está basada en Python, JavaScript y XML, y disponible bajo la licencia AGPLv3.

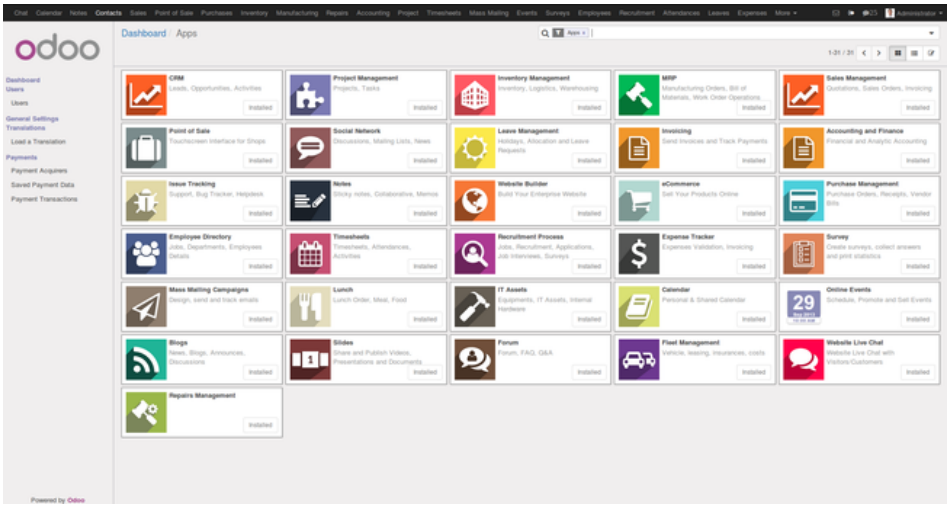


Figura 8: Interfaz de Odoo.
Fuente: `https://demo2.odoo.com/web`

[5] `https://www.odoo.com`

Es posible apreciar lo modular de la solución según la figura 8, la cual separa las tareas de inventario de gestión de proyectos. El módulo de inventario permite operaciones tales como:

- Órdenes de entrega.
- Ajustes de inventario.
- Recibos.
- Transferencias de stocks.
- Informes de revisión.

También permite integración con módulos de contabilidad y ventas, entre otras características[6], así como módulos para eventos, los cuales incluyen conferencias, clases y exposiciones con organización de calendario. Los puntos débiles de esta solución se encuentran en aquellos requerimientos relacionados con las reservas, el manejo de admisión de proyectos y gestión de uso del equipamiento en el Fab Lab se ven comprometidos a un proceso de desarrollo e implementación de funcionalidades no incluídas.

Las ventajas y desventajas que se estiman de esta opción se encuentran en la tabla 2:

Ventajas	Desventajas
Módulos que cumplen algunos requerimientos	Mayor esfuerzo de desarrollo
Menor tiempo de implementación	Mayor esfuerzo operacional

Tabla 2: Ventajas y desventajas de Odoo.
Fuente: Elaboración Propia.

2.2.2. PLATAFORMA INTERNA MEDIANTE PARAMETRIZACIÓN DE TEMPLATES

Una plataforma interna mediante parametrización y modularización de templates consiste en la reutilización de plantillas, código fuente, repositorios y herramientas acequibles que permitan el trabajo unificado en un framework predeterminado y desarrollar el set de requerimientos en éste para cumplir con los objetivos establecidos. Los criterios de selección para las tecnologías a emplear son similares a los expuestos en 1.3.3, bajo los cuales se tienen las siguientes opciones:

- **Ruby on Rails**: RoR es un framework de aplicaciones web de código abierto bajo la licencia MIT. Posee soporte para servidores web, bases de datos, y gemas, las cuales se comportan como librerías con códigos generados por la comunidad

[6] `https://www.odoo.com/es_ES/page/warehouse-features`

que permiten nuevas funcionalidades sin la necesidad de 'reinventar la rueda'. Funciona bajo el patrón de arquitectura MVC, que permite una separación en capas de los datos, las reglas de negocio y la interacción de éstos con el usuario. Entre las funcionalidades que se pueden encontrar realizadas están:

- Autenticación y autorización.
- Dashboards.
- Validaciones locales. Ejemplo: RUN.
- Integración Bootstrap, CSS.

Una desventaja de este framework es que no se adapta de manera rápida a frecuentes cambios en estructura y lógica de datos, por lo que en un entorno como el Fab Lab que no tiene definido procesos estandarizados para sus operaciones, pueden ocurrir cambios inesperados en datos, modelos y diagramas en los que la plataforma se base.

- **OpenXava**: Framework web que permite desarrollo de módulos CRUD y generación de reportes de manera fácil, ya que provee interfaces, acceso de datos, comportamientos por defecto sin necesidad de programarlo. Está orientado a componentes de negocio, lo que implica que todas las definiciones, estructura, mapeo, interfaz, validaciones, cálculos, etc. se encuentran en un solo lugar, favoreciendo la facilidad para actualizar lógica y estructura de datos, así como distribución de tareas basada en la lógica del negocio. Posee licencia GNU LGPL.

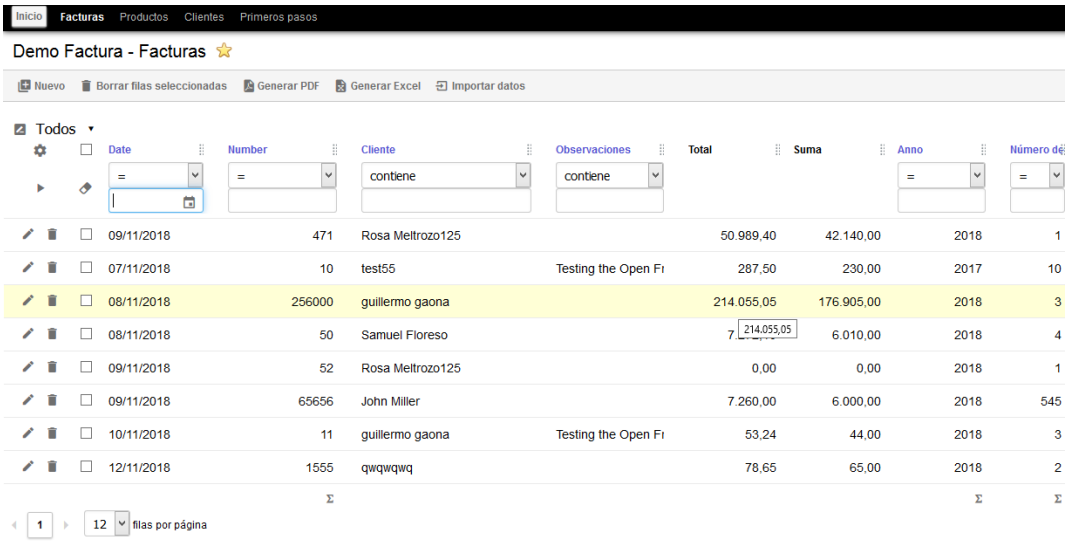


Figura 9: Interfaz esperada con OpenXava.
Fuente: https://openxava.org/demos/

En la tabla 3 se encuentran las ventajas y desventajas de esta propuesta, independiente del framework web a utilizar.

Ventajas	Desventajas
Uso balanceado de tiempo	Mayor esfuerzo de desarrollo
Alto apoyo de la comunidad open source	Falta de experiencia en el framework
Modularización alta	Curva de aprendizaje media

Tabla 3: Ventajas y desventajas de plataforma interna mediante parametrización.
Fuente: Elaboración Propia.

2.2.3. PLATAFORMA INTERNA A MEDIDA

Una Plataforma interna como proyecto in-house a medida, se desarrolla una solución de primeros principios, ocupando de manera exclusiva código fuente, algoritmos e interfaces generados dentro del equipo de trabajo, los cuales deben respaldarse dentro de un repositorio propio. Los criterios de selección para las tecnologías y frameworks web a emplear son los mismos, por lo que es posible utilizar las opciones ya establecidas en la alternativa 2.2.2, teniendo en consideración el desarrollo completo de la solución.

En la tabla 4 se encuentran las ventajas y desventajas de esta propuesta, independiente del framework web a utilizar.

Ventajas	Desventajas
Alto nivel de personalización	Alto esfuerzo y tiempo de desarrollo
Documentación y soporte local	Alto esfuerzo de mantención
Familiaridad con código generado	

Tabla 4: Ventajas y desventajas de Proyecto Interno In-house.
Fuente: Elaboración Propia

2.3. CRITERIOS DE SELECCIÓN

Para determinar la alternativa a utilizar, se tienen en consideración factores tanto beneficiosos como riesgosos que impactan negativamente al proyecto (designados con el símbolo '+' o '-' respectivamente). Los factores son los siguientes:

- Esfuerzo de desarrollo (-): Estimación del costo en tiempo HH de desarrollo. A mayor costo, más riesgo.
- Esfuerzo de soporte (-): Estimación del costo de dedicación a resolución de problemas. A mayor costo, más riesgo.
- Esfuerzo de mantención (-): Estimación de costo para nuevas necesidades o mejoras. A mayor costo, más riesgo.

- Cumplimiento de funcionalidades (+): Medición de cumplimiento de funcionalidades. A mayor realización, mayor beneficio.

Con una escala de 1 a 5, representando intensidad de beneficio o riesgo dependiendo del criterio (A mayor valor, mayor riesgo o beneficio). Para asignar valor, se tendrán en consideración las prioridades de los stakeholders en entrevistas, así como las fortalezas y debilidades de quien lleve a cabo el proyecto, tal que se establecen en consenso para cada alternativa, calculando un promedio aritmético para los beneficios y para los riesgos separadamente. La mejor alternativa debe acercar los factores positivos a 5 y los negativos a 1.

Si se utiliza una plataforma externa, facilita la muestra de un producto probado, con requerimientos parcialmente listos, posibilitando una implementación y prueba rápida de la plataforma. Sin embargo, uno de los puntos importantes para los stakeholders es la capacidad de generar conocimiento a través del análisis del uso de recursos dentro del Fab Lab, aspecto que está ausente en esta solución.

En la tabla 5 se presenta una evaluación del aspecto beneficioso de esta opción.

Criterio	Puntaje
Cumplimiento de requerimientos	2.5

Tabla 5: Evaluación de aspectos beneficiosos para plataforma externa.
Fuente: Elaboración Propia

En la tabla 6 se presenta una evaluación de los aspectos riesgosos de esta opción.

Criterio	Puntaje
Esfuerzo de desarrollo	1
Esfuerzo de soporte	3
Esfuerzo de mantención	4

Tabla 6: Evaluación de aspectos riesgosos para plataforma externa.
Fuente: Elaboración Propia

Si la opción que se escogiera fuera una plataforma interna con posibilidad de parametrizar y modularizar templates existentes, es posible reutilizar código fuente, lo cual resulta en un ahorro de tiempo y esfuerzo de desarrollo para cumplir los requerimientos solicitados. Esto trae otra ventaja: La redistribución de tiempo y esfuerzo en nuevas funcionalidades y la mejora con las que se desarrollen. El cumplimiento de requerimientos puede adaptarse a las necesidades del Fab Lab.

En la tabla 7 se presenta una evaluación del aspecto beneficioso de esta opción.

Criterio	Puntaje
Cumplimiento de requerimientos	4

Tabla 7: Evaluación de aspectos beneficiosos para plataforma externa mediante templates.
Fuente: Elaboración Propia

En la tabla 8 se presenta una evaluación de los aspectos riesgosos de esta opción.

Criterio	Puntaje
Esfuerzo de desarrollo	2
Esfuerzo de soporte	3
Esfuerzo de mantención	3

Tabla 8: Evaluación de aspectos riesgosos para plataforma externa mediante templates.
Fuente: Elaboración Propia

En caso que se optara por una plataforma interna como un proyecto in-house a medida, es posible asegurar el cumplimiento completo de los requerimientos con según las necesidades específicas del Fab Lab. La mayor desventaja que presenta es la inversión de tiempo y esfuerzo en desarrollar cada aspecto de la plataforma, el cual involucra esencialmente código fuente, optimización de algoritmos, documentación y personal capacitado.

En la tabla 9 se presenta una evaluación del aspecto beneficioso de esta opción.

Criterio	Puntaje
Cumplimiento de requerimientos	5

Tabla 9: Evaluación de aspectos beneficiosos para plataforma externa hecha a medida.
Fuente: Elaboración Propia

En la tabla 10 se presenta una evaluación de los aspectos riesgosos de esta opción.

Criterio	Puntaje
Esfuerzo de desarrollo	5
Esfuerzo de soporte	3
Esfuerzo de mantención	4

Tabla 10: Evaluación de aspectos riesgosos para plataforma externa hecha a medida.
Fuente: Elaboración Propia

En resumen, los promedios aritméticos de beneficios y riesgos para las alternativas se presentan en la tabla 11.

Propuesta	Promedio (+)	Promedio (-)
Plataforma externa	2.5	2.7
Plataforma interna mediante templates	4	2.7
Plataforma interna a medida	5	4

Tabla 11: Promedio de aspectos positivos y negativos evaluados para cada solución.
Fuente: Elaboración Propia

En base a la información expuesta, se escoge desarrollar la plataforma de manera interna mediante parametrización y modularización de templates detallado en la sección 2.2.2, ya que la evaluación de esta solución respecto a las otras alternativas determina que maximiza el cumplimiento de requerimientos y minimiza los aspectos riesgosos relacionados con el esfuerzo y tiempo a invertir en la construcción de esta plataforma.

2.4. METOLODOGÍA DE DESARROLLO

Se propone utilizar método SCRUM como base para planificación del desarrollo para otorgar avances o entregables incrementales que puedan recibir retroalimentación y documentación. Las razones por las cuales se escoge este método de trabajo son las siguientes:

- Los requisitos funcionales expresados en la sección 3.2 permiten desarrollar bases mínimas las cuales pueden ser modificadas y aumentadas de manera excluyente.
- Las entregas periódicas facilitan la entrega de retroalimentación por parte del cliente, por lo que es posible consolidar funcionalidades en menor tiempo.
- El método se adapta ante cambios regulares en las necesidades.

Los roles que los distintos stakeholders toman son los siguientes:

- Scrum Manager: Aquella persona más cercana a los roles de análisis y diseño del sistema debe tomar este rol, organizar el equipo y actuar como mediador entre el equipo de desarrollo y el Product Owner supervisando los requerimientos y asesorando las especificaciones.
- Product Owner: Lab Administrator toma este rol para focalizar al equipo en los aspectos del negocio principales.
- Equipo técnico: El grupo de desarrolladores que dedique sus esfuerzos a avanzar el proyecto periódicamente. Deben conocer los requerimientos que el Scrum Manager otorgue.

Se estiman sprint semanales, donde se desarollan reuniones de inicio para elaborar un Sprint Backlog con las tareas, requisitos y condiciones bajo las cuales se va a realizar en el período del sprint, de índole técnica por lo menos en 2 ocasiones por sprint y de finalización, donde se revisan, evalúan y mejoran las tareas realizadas, los métodos de realización y el producto avanzado.

Descripción	Costo [Aprox.]	Unidad
Esfuerzo de desarrollo	166	HH
Esfuerzo de soporte	56	HH
Esfuerzo de mantención	56	HH

Tabla 12: Costos de la solución desarrollada en RoR.
Fuente: Elaboración Propia.

A continuación se presenta una propuesta de duraciones de desarrollo que tiene en consideración la elaboración concreta de requisitos funcionales y la documentación respectiva en la figura 13.

ID	Nombre Tarea	Duración [HH]
1	Tareas de coordinación y gestión iniciales	
1.1	Reuniones de coordinación	4
1.2	Capacitación en Herramientas de Desarrollo	16
2	Implementación de Base de Datos	
2.1	Generación de Modelos en DMT	16
2.2	Generación de Reportes y Scripts de Respaldo	8
2.3	Pruebas de conexión y datos	8
2.4	Documento de Pruebas	4
2.5	Documento de Revisión y Ajustes	4
3	Funciones Preliminares	
3.1	Creación de Cuenta	8
3.2	Implementación de Roles	16
3.3	Datos de Prueba	8
3.4	Documento de Pruebas	4
3.5	Documento de Revisión y Ajustes	4
4	Admisión de Proyectos	
4.1	Crear Proyecto	8
4.2	Unirse a Proyecto	8
4.3	Gestionar Proyecto	8
4.4	Evaluar Proyecto	16
4.5	Documento de Pruebas	4
4.6	Documento de Revisión y Ajustes	4
5	Gestión de Inventario	
5.1	Crear Equipamiento	8
5.2	Crear Insumo	8
5.3	Crear Espacio	8
5.4	Gestionar Equipamiento	8
5.5	Crear Insumo	8
5.6	Gestionar Espacio	8
5.7	Documento de Pruebas	4
5.8	Documento de Revisión y Ajustes	4
6	Administración de Reservas	
6.1	Efectuar Reserva	32
6.2	Evaluar Reserva	16
6.3	Documento de Pruebas	8
6.4	Documento de Revisión y Ajustes	8
7	Finalización	
7.1	Reunión de Entrega	4
	Total Duración	138

Tabla 13: Planificación de desarrollo para plataforma.
Fuente: Elaboración Propia

2.5. GESTIÓN DE RIESGOS

A continuación se presentan riesgos inherentes al desarrollo y las circunstancias bajo las cuales se planea desarrollar el proyecto.

Desde la tabla 14 hasta la tabla 20 contienen en detalle cómo se aborda cada riesgo identificado, indicando la prioridad, impacto, probabilidad, tipo de riesgo, contexto, plan de contingencia y de mitigación para cada uno de ellos.

La cuantificación asociada a cada detalle es:

- *Prioridad e Impacto*: entre más cercano a 1 es menor (escala de 1 a 5).
- *Probabilidad*: entre más cercano a 1 es mayor. (escala de 0 a 1).

Riesgo RT1		Dependencia del internet de la universidad para plataforma web.			
Prioridad	3	**Impacto**	4	**Probabilidad**	0.5
Tipo de Riesgo		Técnico.			
Contexto		Se ha pedido una plataforma web que dependerá de los servicios de internet de la universidad para funcionar. La falla de este servicio significaría no disponibilidad de la plataforma.			
Plan de Contingencia		Tener una segunda fuente de servicio de internet. También es posible efectuar respaldos físicos de información relevante de manera periódica para operar de manera off-line.			
Plan de Mitigación		Efectuar documentación física de las operaciones que se realicen hasta la reanudación del servicio, para luego subir esta información actualizada.			

Tabla 14: Control de Riesgo Técnico: RT1

Riesgo RT2		Escalabilidad insuficiente.			
Prioridad	3	**Impacto**	3	**Probabilidad**	0.4
Tipo de Riesgo		Técnico.			
Contexto		Esta fase del proyecto es la base para futuros requerimientos que se necesiten. Una falta de escalabilidad significaría, en el peor caso, iniciar un plan de desarrollo para una nueva plataforma que lo reemplace.			
Plan de Contingencia		Escoger herramientas de trabajo que favorezcan integración, escalabilidad y una arquitectura modular.			
Plan de Mitigación		Poseer respaldos del desarrollo de la plataforma que permita efectuar rollback a requerimientos y optimizar el reuso de código útil.			

Tabla 15: Control de Riesgo Técnico: RT2

Riesgo RPr1		Subestimación del esfuerzo y tiempo requerido.			
Prioridad	4	**Impacto**	4	**Probabilidad**	0.7
Tipo de Riesgo		Proyecto.			
Contexto		Una mala estimación dada por la poca experiencia en planificación de proyectos puede causar un atraso en los plazos o incumplimiento en los objetivos.			
Plan de Contingencia		Agendar reuniones periódicas con stakeholders generar instancias de avance y retroalimentación. Tener asesoría externa que permita direccionar mejor la productividad así como el tecnicismo de los resultados obtenidos.			
Plan de Mitigación		Reprogramar plazos agregando no más del 12,5 % del tiempo inicial.			

Tabla 16: Control de Riesgo de Proyecto: RPr1

Riesgo RPr2		Poca disponibilidad del cliente para reuniones de feedback.			
Prioridad	4	**Impacto**	3	**Probabilidad**	0.7
Tipo de Riesgo		Proyecto.			
Contexto		Debido al cómo se desarrollan los procesos actualmente, es posible que el clientes no esté disponible de manera periódica para encabezar la revisión el progreso de la plataforma, causando incertidumbre en lo acertado del progreso que se lleve.			
Plan de Contingencia		Aprovechar las primeras reuniones en obtener la mayor cantidad de información relevante que permita un mayor avance independiente.			
Plan de Mitigación		Facilitar la entrega de información por medios remotos: videoconferencia, documentación online, e-mail, entre otros.			

Tabla 17: Control de Riesgo de Proyecto: RPr2

Riesgo RPr3		Personal insuficiente para completar los requerimientos.			
Prioridad	2	**Impacto**	3	**Probabilidad**	0.4
Tipo de Riesgo		Proyecto.			
Contexto		El alcance del proyecto a largo plazo tiene un amplio espectro, y actualmente sólo se lleva a cabo por una persona, por lo que múltiples tareas de diferentes ámbitos tendrá que ser hecho por una misma persona, repercutiendo en el tiempo y calidad del proyecto, y también ejerce mayor presión y estrés en la persona encargada.			
Plan de Contingencia		Reclutar personas dentro de la comunidad del Fab Lab que, según sus habilidades, puedan ejercer labores que apoyen al proyecto en ámbitos como la gestión y la documentación, así el desarrollador puede enfocar sus esfuerzos en desarrollar.			
Plan de Mitigación		Ampliar plazos de realización para cada etapa del proyecto en no más de un 12,5 % del establecido inicialmente y establecer nuevos plazos.			

Tabla 18: Control de Riesgo de Proyecto: RPr3

Riesgo RPe1		Poca experiencia en las herramientas a utilizar.			
Prioridad	4	**Impacto**	4	**Probabilidad**	0.7
Tipo de Riesgo		Personal.			
Contexto		Los parámetros específicos al momento de establecer una solución requiere la utilización de herramientas que faciliten agregar funcionalidad e integración de personal, con lo que lleva a soluciones con las cuales no necesariamente existe experiencia.			
Plan de Contingencia		Considerar en el tiempo total y en la planificación un tiempo de capacitación en las herramientas a utilizar. Esto puede implicar contratar personal.			
Plan de Mitigación		Invertir en personal que ya posea los conocimientos y la experiencia necesaria para llevar el proyecto en los plazos establecidos.			

Tabla 19: Control de Riesgo Personal: RPe1

Riesgo RPe2		Inconvenientes de salud o personales de otro ámbito.			
Prioridad	2	**Impacto**	3	**Probabilidad**	0.25
Tipo de Riesgo		Personal.			
Contexto		Eventos fortuitos tales como complicaciones de salud de los desarrolladores, luto por muerte de un ser querido, actividades laborales o extracurriculares, entre otros, pueden retrasar el desarrollo del proyecto e incumplimiento de la planificación inicial.			
Plan de Contingencia		Desarrollo intenso del proyecto en una etapa inicial que permita mayor flexibilidad posterior. Asignar roles de reemplazo o secundarios si se dispone de personal.			
Plan de Mitigación		Se redefinen los recursos encargados de las tareas dentro de los avances, priorizando aquellas que permitan una mayor tasa de entregables y una mayor facilidad de desarrollo.			

Tabla 20: Control de Riesgo Personal: RPe2

2.6. HERRAMIENTAS DE SOPORTE PARA EL DESARROLLO

Se han considerado las siguientes herramientas y plataformas de desarrollo para el proyecto FLUSM:

- **Ruby on Rails**: Framework de aplicaciones web de código abierto. El motivo por el que se escoge es su facilidad de configuración e integración con las diferentes piezas que componen la plataforma: Soporte para servidores web, bases de datos, y gemas, las cuales se comportan como librerías con códigos generados por la comunidad que permiten nuevas funcionalidades sin la necesidad de 'reinventar la rueda'. Funciona bajo el patrón de arquitectura MVC, que permite una separación en capas de los datos, las reglas de negocio y la interacción de éstos con el usuario.

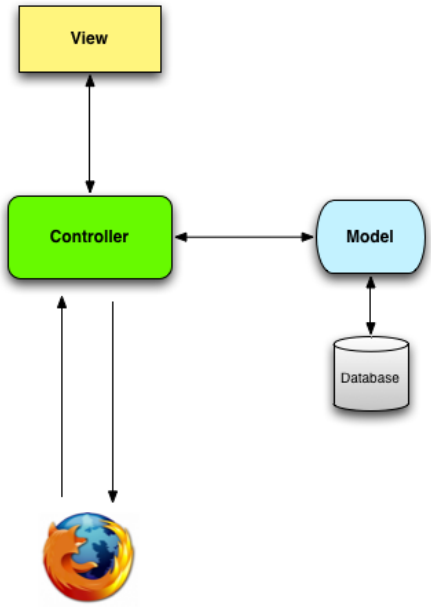


Figura 10: Diagrama de funcionamiento de RoR bajo el esquema MVC.
Fuente: Ruby on Rails Tutorial by Michael Hartl.

- **Git**: Software de control de versiones para una mejor mantención de muchos archivos de código fuente entre múltiples colaboradores. Es recomendado el uso de esta herramienta para mejor control de cambios, y prevenir errores costosos, tales como la eliminación accidental de archivos esenciales. En este informe, se ocupa la plataforma GitHub y BitBucket.
- **Sharelatex, Overleaf**: Editor de documentos en formato LaTeXque permite compilación y trabajo colaborativo en línea y a tiempo real para generación de documentación relevante.

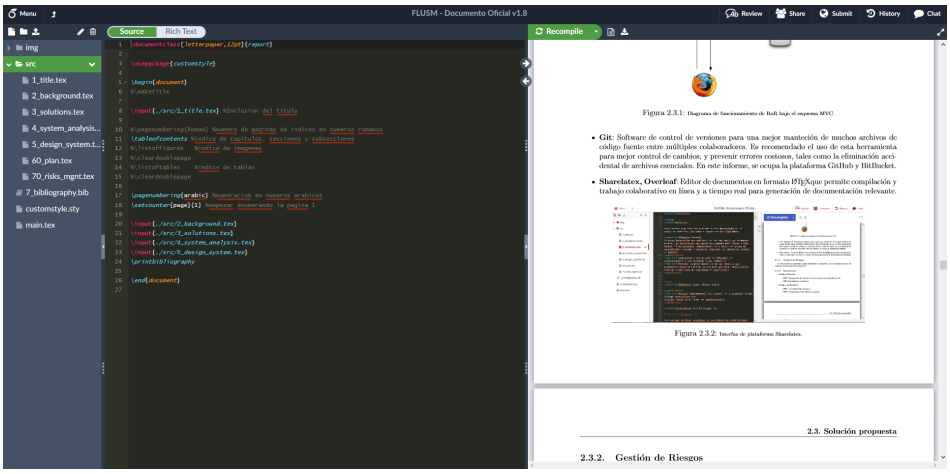


Figura 11: Interfaz de plataforma Overleaf, desde su unión con Sharelatex.
Fuente: Elaboración Propia.

- **Bizagi Modeler**: Aplicación para diagramar, documentar y simular procesos en notación BPMN que permitan entender mejor el desarrollo de los procesos.

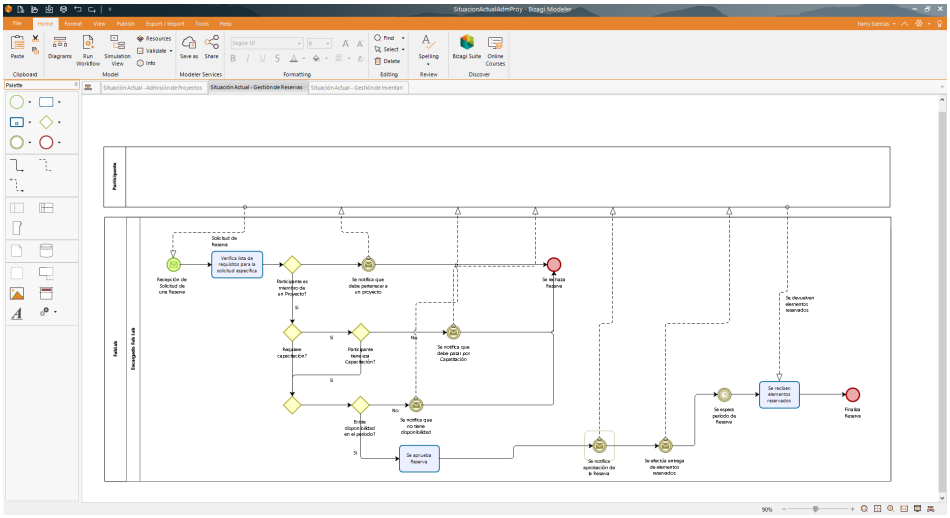


Figura 12: Interfaz de Bizagi Modeler.
Fuente: Elaboración Propia.

CAPÍTULO 3
ANÁLISIS DEL SISTEMA

Este capítulo explica en detalle las funciones que se esperan del sistema a desarrollar y los procesos que se llevan a cabo actualizados teniendo en consideración la plataforma.

3.1. ASPECTOS GENERALES

Se utilizan las siguientes notaciones para mostrar los diferentes procesos y diagramas:

- BPMN: Notación gráfica para modelado de procesos. Con este formato, se busca documentar los procesos existentes en el Fab Lab, así como estandarizar los procesos que involucran a la plataforma a desarrollar
- UML: Lenguaje de modelado de sistemas de software. Se ocupa para efectuar diagramas estructurales y de comportamiento del sistema, tales como diagrama de estados, diagrama de clases, entre otros.

Se identifican cinco roles principales:

- **Usuario No Registrado**: Usuario que no ha ingresado sus datos a la plataforma del Fab Lab.
- **Participante**: Aquellos usuarios registrados en la plataforma quienes pueden ingresar a proyectos existentes. Siendo miembros de un proyecto, les es permitido efectuar reservas en pos del cumplimiento de estos proyectos, para generar resultados comprobables que puedan registrar en el sistema.
- **Jefe de Proyecto**: Aquellos usuarios registrados en la plataforma quienes crean sus propios proyectos, siendo responsables deéste. Les es permitido efectuar reservas en pos del cumplimiento de estos proyectos, para generar resultados comprobables que puedan registrar en el sistema.
- **Operador**: Encargado del proceso de admisión de proyectos, aprobación de reservas y reporte de mediciones en uso del Fab Lab.
- **Bodeguero**: Encargado de la administración del equipamiento y espacios.

Se establece un modelo de dominio, el cual se muestra en la figura 13, basados en los conceptos que dentro del vocabulario utilizado en el Fab Lab se consideran esenciales.

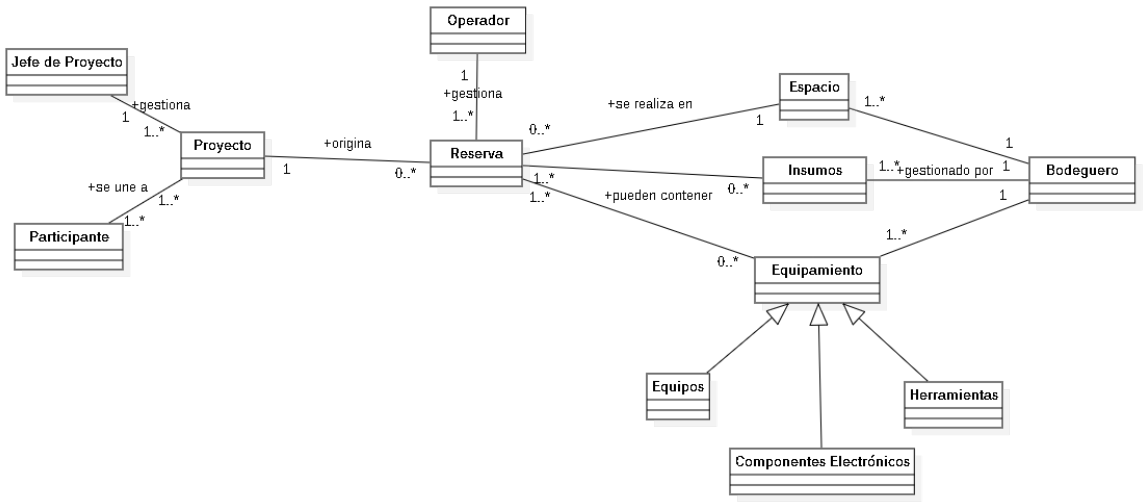


Figura 13: Modelo de Dominio de Plataforma de Manejo de Recursos Fab Lab.
Fuente: Elaboración Propia

Las reservas se encuentran asociadas y justificadas por la existencia un proyecto en desarrollo, ambos elementos siendo revisados y aprobados por un operador. Los elementos que se pueden reservar están categorizados por: Espacio, insumos y equipamiento.

3.2. REQUISITOS DEL SISTEMA

A continuación se presenta un resumen de los requisitos pedidos por el cliente.

3.2.1. REQUISITOS FUNCIONALES

Un usuario no registrado puede efectuar las siguientes tareas en la plataforma:

- **RF-UNR01**: Crear cuenta.

Un jefe de proyecto debe poder efectuar las siguientes tareas en la plataforma:

- **RF-J01**: Crear, editar y eliminar proyectos propios.
- **RF-J02**: Permitir editar o eliminar miembros de un proyecto propio.
- **RF-J03**: Ver estadísticas de su actividad en plataforma: proyectos creados, cargos efectuados y reservas solicitadas.

Un participante debe poder efectuar las siguientes tareas en la plataforma:

- **RF-P01**: Postular a proyectos existentes.
- **RF-P02**: Participante que sea miembro de un proyecto puede gestionar reservas que efectúe para el avance de ese proyecto. O sea, debe permitir crear, modificar y eliminar sus reservas.
- **RF-P03**: Ver estadísticas de su actividad en la plataforma: Proyectos a los que pertenece, cargos ejercidos y reservas realizadas.

Un operador debe poder efectuar las siguientes tareas en la plataforma:

- **RF-O01**: Debe permitir aprobar o rechazar una solicitud de proyecto en base a la visualización de criterios predefinidos.
- **RF-O02**: Editar, aprobar y deshabilitar reservas de cualquier proyecto.
- **RF-O03**: Visualizar estadísticas relacionadas con el uso y los costos de utilización del Fab Lab, los proyectos que se inscriben y las reservas que realizan.

Un bodeguero debe poder efectuar las siguientes tareas en la plataforma:

- **RF-B01**: Agregar, editar y deshabilitar elementos en Inventario.
- **RF-B02**: Agregar, editar y deshabilitar espacios.
- **RF-B03**: Ver estadísticas relacionadas con el equipamiento, insumos y espacios: cantidades disponibles e historial de estados.

3.2.2. REQUISITOS NO FUNCIONALES

- **RNF-01**: El nuevo sistema se acoge a las reglas de las licencias generales públicas (GNU) de los códigos que utilice. Vale decir, en términos generales, que los códigos fuente generados son abiertos y gratuitos, en el que cualquiera puede cambiar el software, sin patentes y sin garantías.
- **RNF-02**: La aplicación web debe poseer un diseño que garantice la adecuada visualización en múltiples computadores personales, dispositivos tableta y teléfonos inteligentes.

3.2.3. FUERA DE ALCANCE

Los siguientes elementos no son considerados al momento de analizar requerimientos:

- **Finanzas**: Aspectos que involucren cuantificar, medir y analizar la realidad económica del Fab Lab. Esto implica que quedan fuera reportes financieros, manejos y propuestas de presupuestos, cotizaciones, compras y evaluaciones económicas en general.
- **Planificación de Proyectos**: Todo lo relacionado con la planificación, desarrollo y ejecución de los proyectos que se lleven a cabo en el Fab Lab se asumen a cargo de los propios miembros del equipo a cargo, con lo que esta iniciativa no otorga herramientas que puedan ser utilizadas con estos propósitos.
- **Gestión de Capacitaciones**: El registro, almacenamiento y reportabilidad de la planificación, ejecución y asistencia de talleres, capacitaciones y otros eventos efectuados por el personal del Fab Lab, aparte de las reservas en si, son funcionalidades no consideradas en este proyecto.
- **Migración de Datos**: Tareas como la extracción, transformación y carga de datos desde sus fuentes originales a las bases de datos finales no son tratadas en este documento.

3.3. NUEVOS PROCESOS

Como se determina en la sección 'Aspectos a Mejorar' en la sección 1.2.4, es necesaria la definición de procesos actualizados que involucren a la plataforma.

Se consideran tres procesos necesarios a analizar y mejorar:

- **Admisión de Proyectos**: Es la primera barrera de entrada al Fab Lab que procura un uso práctico de sus recursos en pos del desarrollo de proyectos.
- **Administración de Reservas**: El vínculo principal entre los proyectos con los que el Fab Lab busca cooperar y los recursos que han podido adquirir. Estructurar este proceso permite maximizar los quehaceres del Fab Lab y de los proyectos.
- **Gestión de Inventario**: Proceso menester para la ejecución efectiva de las reservas y mantener un mejor control de lo que se tiene dentro del Fab Lab, se busca identificar mejor las transacciones que se efectúen y obtener información más objetiva del uso de los elementos.

Para explicar estos procesos, se utilizará notación BPMN, cuya simbología se encuentra detallada en la sección 5.1.

3.3.1. ADMISIÓN DE PROYECTOS

El objetivo con admisión de proyecto es migrar la información existente y la información nueva a generar en datos acequibles, manipulables y que se orienten al análisis para la toma de decisiones. Se requiere que aquellos participantes con solicitudes de proyectos, previamente se registren en la plataforma. Con esto se busca tener mayor control de los integrantes que conforman los proyectos en el Fab Lab. Como paso esencial para los participantes, es necesario que exista una evaluación que incluya criterios de recepción de proyectos que sean de interés para el Fab Lab, de tal manera que permita priorizar o descartar proyectos acorde a sus necesidades. Los criterios que se encuentran presentes en sus formularios o son importantes para el Fab Lab son los siguientes:

- **Open Knowledge**: Proyecto posee componentes, piezas, documentación o contenido de software o hardware con licencia de código abierto u otro licenciamiento que permita o incentive 'Open Knowledge'.
- **Colaboración**: Proyecto presenta relación con la comunidad USM, o considera voluntariados al interior del Fab Lab por parte de los integrantes del team que efectúan la idea.
- **Financiamiento**: Proyecto tiene aprobado fondos concursables, becas o alguna clase de financiamiento externo.

A estos criterios se les asigna un puntaje que permita valorizar el cumplimiento a juicio del operador. El esquema presentado en la figura 14 se presenta el proceso esperado con la plataforma, con las nuevas tareas requeridas encerradas en un rectángulo verde.

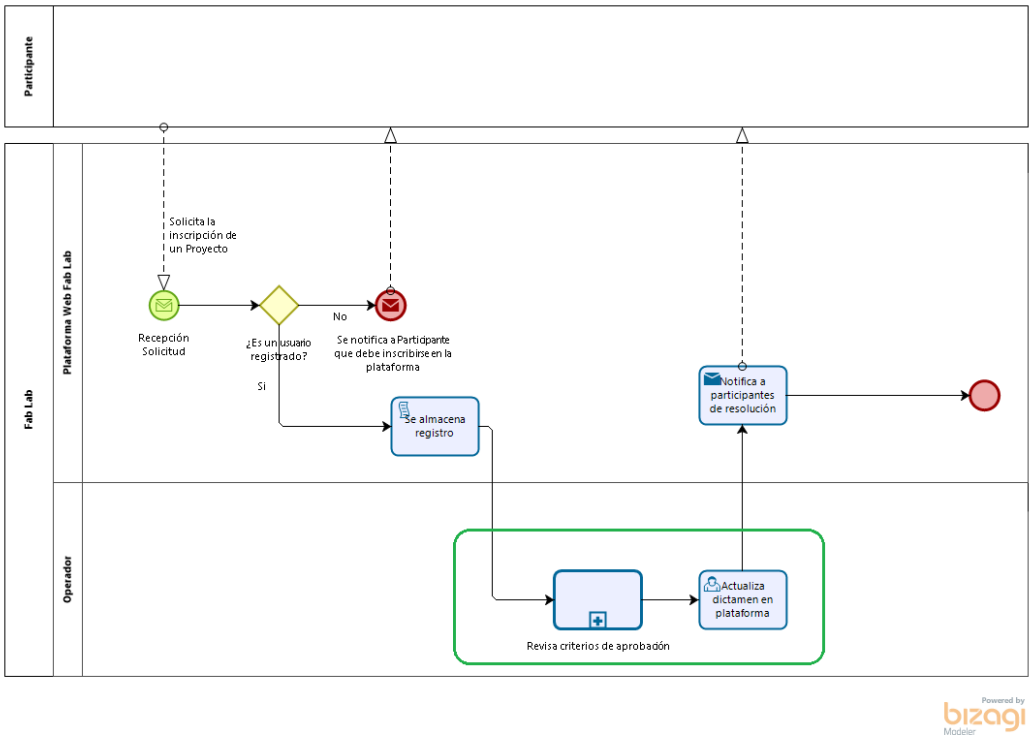


Figura 14: Diagrama de Proceso Admisión de Proyectos con Plataforma Web.
Fuente: Elaboración Propia.

3.3.2. ADMINISTRACIÓN DE RESERVAS

Uno de los problemas más relevantes que tiene el Fab Lab actualmente es la ausencia de una metodología de registro, almacenamiento y estudio de todos los datos relevantes en una reserva. La idea fundamental es que la plataforma tome un rol protagónico en estos aspectos desde el registro de la reserva hasta el posterior estudio del comportamiento de los proyectos con las reservas mediante reportes, tareas que asume el operador en el nuevo sistema. Se define el proceso con la definición de tareas en la figura 15.

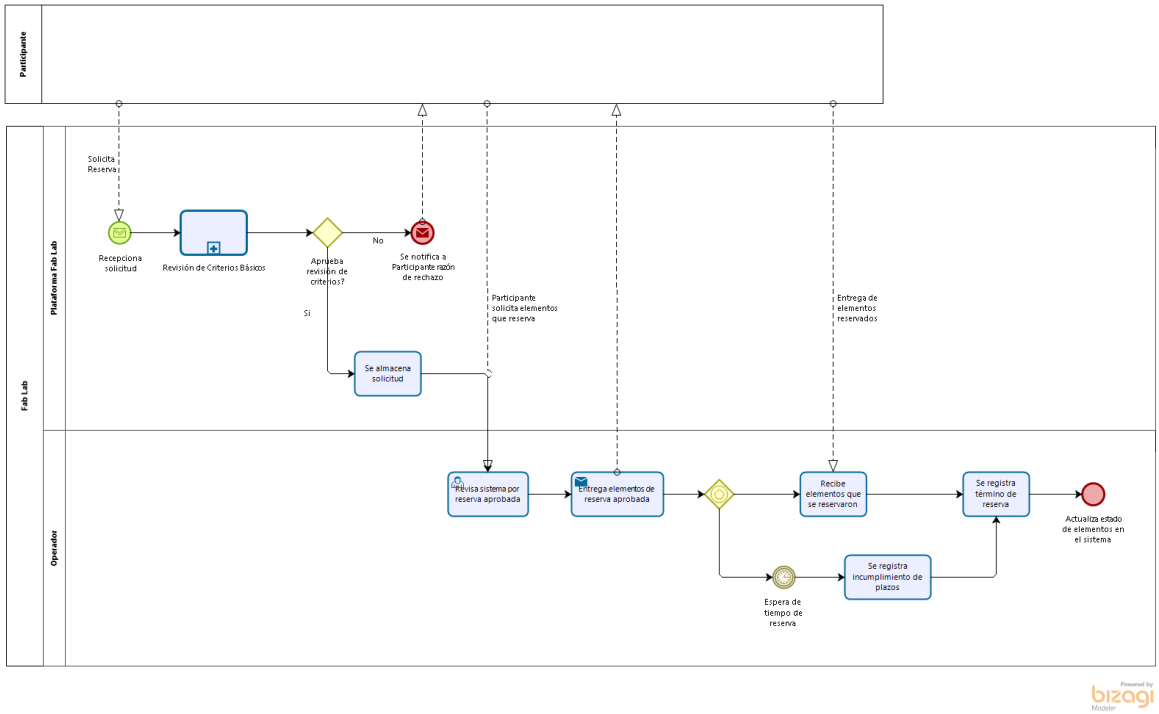


Figura 15: Diagrama de Proceso Administración de Reservas con Plataforma Web.
Fuente: Elaboración Propia

Se modifica la manera en que se llevan los criterios de selección. La plataforma es la encargada de almacenar los datos registrados por los miembros del proyecto, dejar la visualización de estos criterios al operador, la actualización del estado de la reserva, así como archivar posibles faltas cometidas por el equipo que lleva a cabo el proyecto, tales como: incumplimiento de plazos de entrega de elementos, entrega de elementos en mal estado, uso indebido de elementos reservados, entre otros.

Una reserva queda solicitada cuando un participante la registra en el sistema. Luego las reservas se someten a la aprobación de la plataforma, que determina según condiciones básicas, tales como disponibilidad, estado del proyecto de quien reserva y, dependiendo del elemento a reservar, requerimientos de asistencia a talleres de capacitación para que pueda ser aprobada o rechazada. En la figura 16 se muestra el resumen de esta iteración de cumplimiento de criterios.

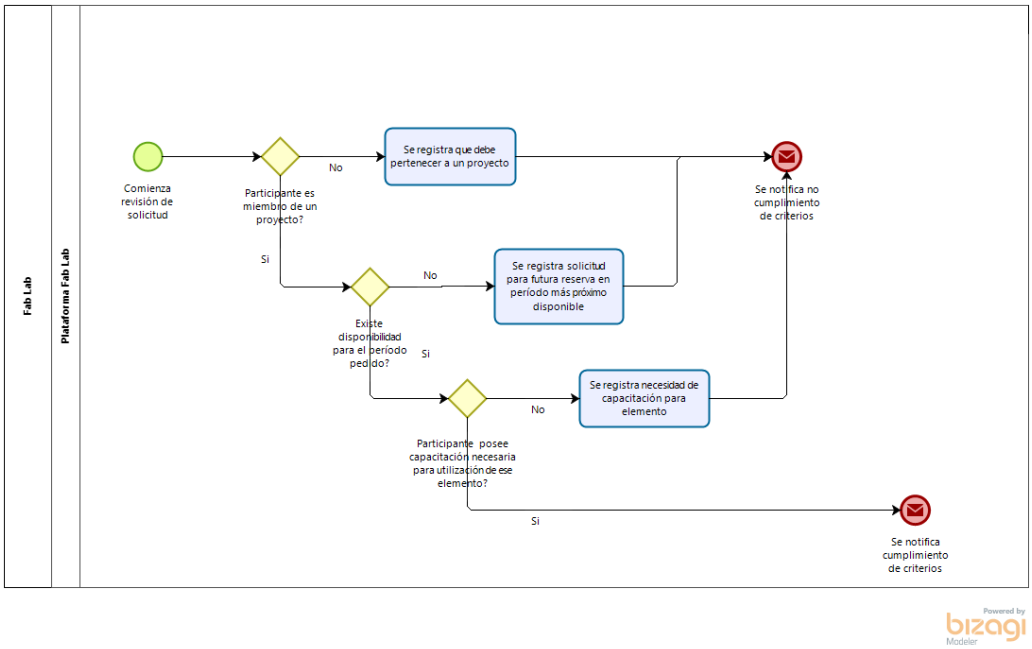
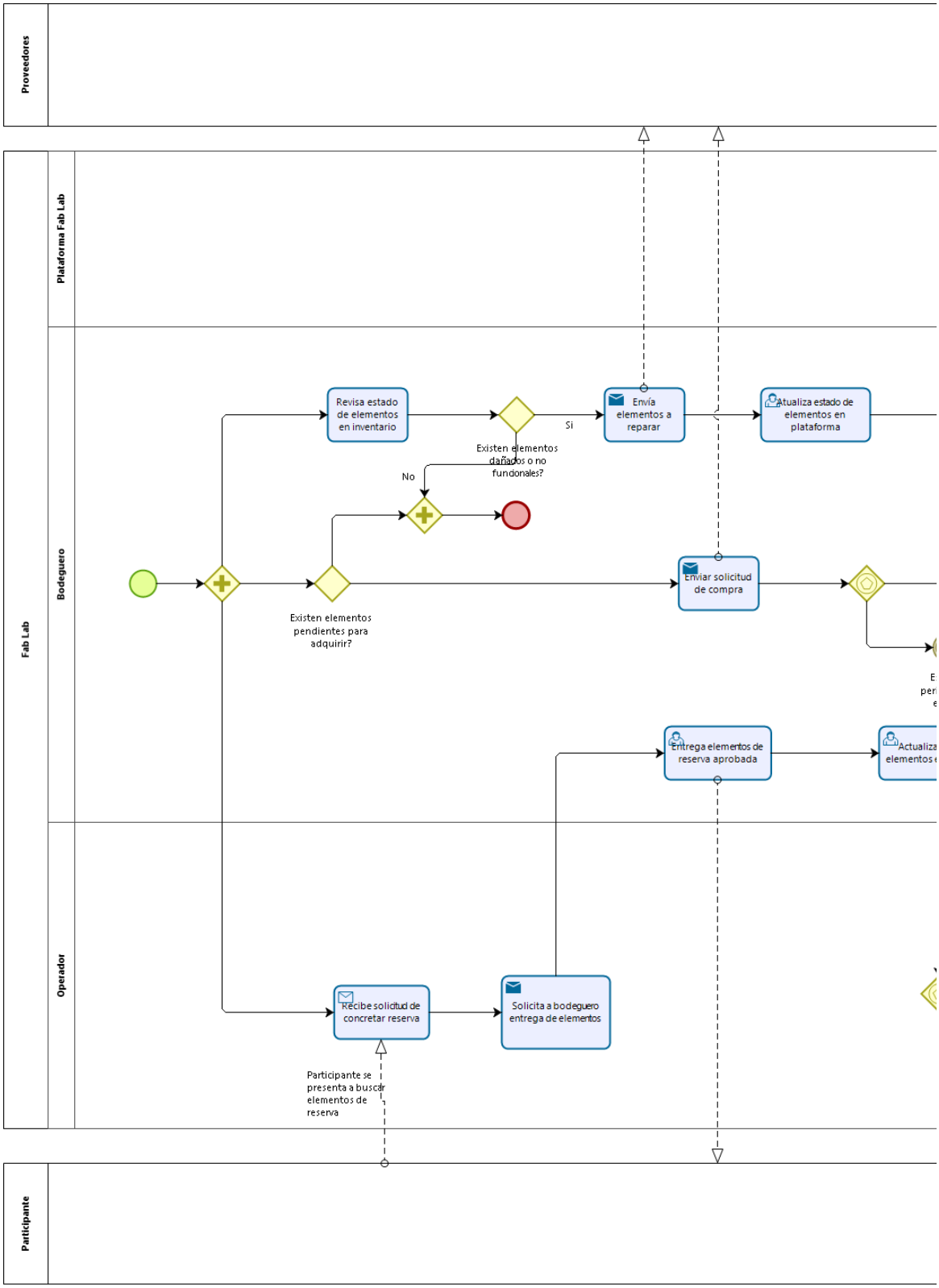
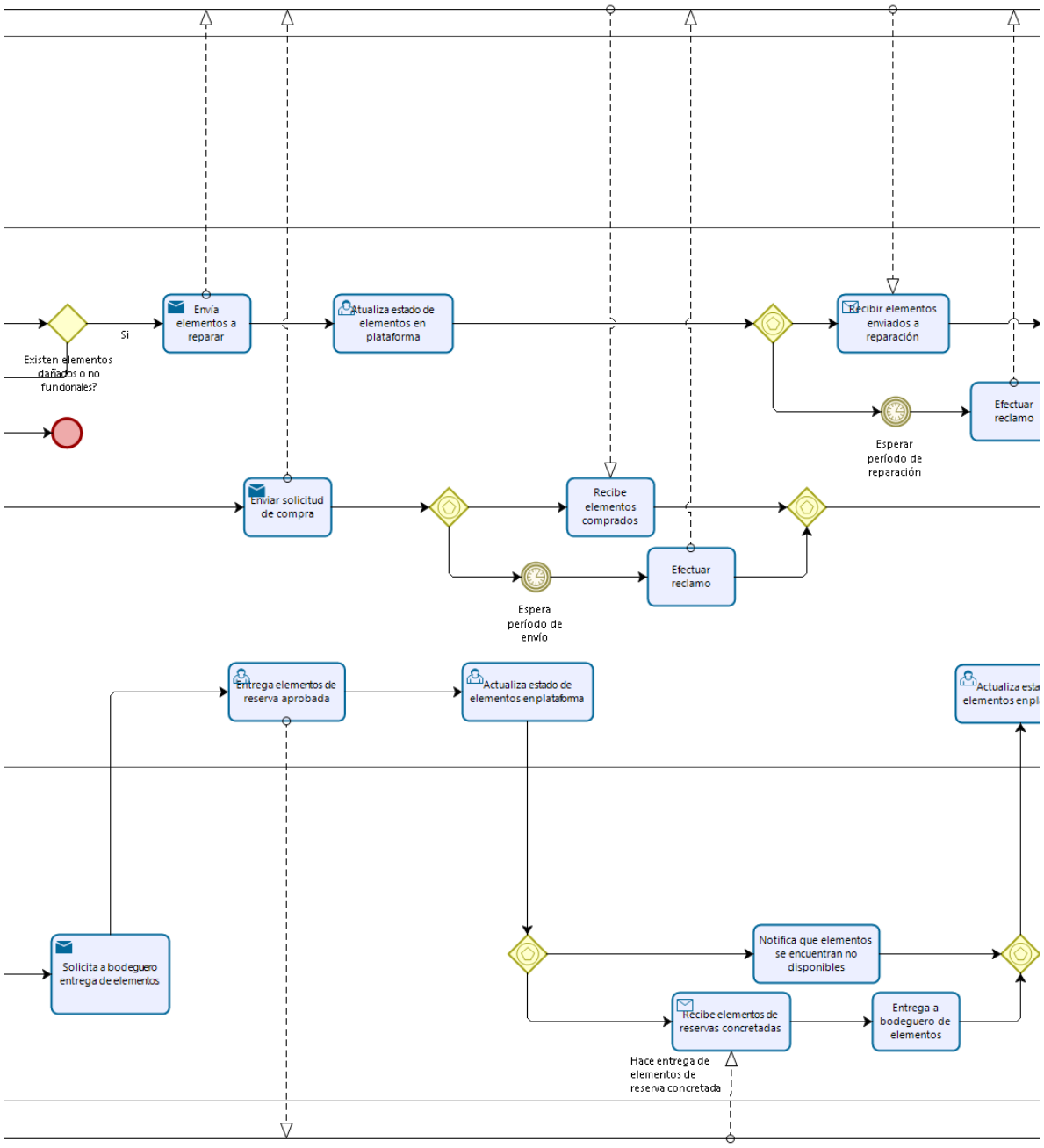


Figura 16: Diagrama de Subproceso de Aprobación de Criterios para Aprobar Reserva. Fuente: Elaboración Propia

3.3.3. GESTIÓN DE INVENTARIO

El equipamiento consiste en una serie de elementos que se clasifican en primera instancia dentro de las siguientes categorías en base a lo expresado en la fig. 13: Equipos, Componentes Electrónicos y Herramientas. Está a cargo del bodeguero, quien designa la disponibilidad en cuanto ingresa los elementos al sistema. También actualiza el estado de cada elemento según necesidades de reparación o actualización bajo los tres eventos que se designan como causantes de cambio de estado: Adquisicón de productos, reparación de productos o reservas a concretar.


Proveedores
Fab Lab
Plataforma Fab Lab
Bodeguero
Operador
Participante
Revisa estado de elementos en inventario
Existen elementos dañados o no funcionales?
Si
No
Envía elementos a reparar
Atualiza estado de elementos en plataforma
Existen elementos pendientes para adquirir?
Enviar solicitud de compra
Entrega elementos de reserva aprobada
Recibe solicitud de concretar reserva
Solicita a bodeguero entrega de elementos
Participante se presenta a buscar elementos de reserva


Existen elementos dañados o no funcionales?
Si
Envía elementos a reparar
Atualiza estado de elementos en plataforma
Recibir elementos enviados a reparación
Efectuar reclamo
Esperar período de reparación
Enviar solicitud de compra
Recibe elementos comprados
Efectuar reclamo
Espera período de envío
Entrega elementos de reserva aprobada
Actualiza estado de elementos en plataforma
Solicita a bodeguero entrega de elementos
Notifica que elementos se encuentran no disponibles
Recibe elementos de reservas concretadas
Entrega a bodeguero de elementos
Hace entrega de elementos de reserva concretada

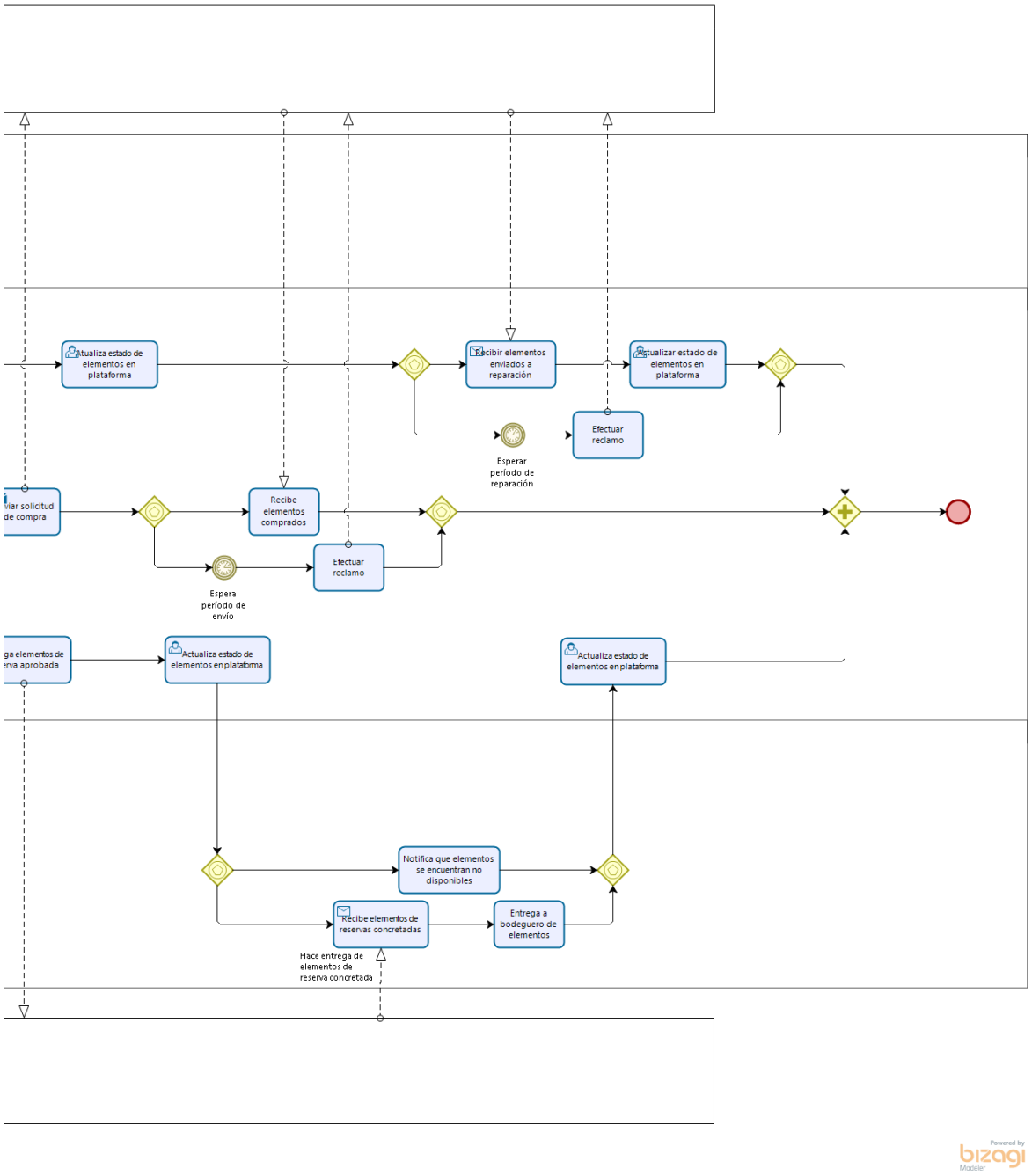


Figura 17: Diagrama de Proceso Gestión de Inventario con Plataforma Web.
Fuente: Elaboración Propia

3.4. CASOS DE USO

El sistema debe permitir crear cuenta a cualquier persona no registrada en el sistema, sin embargo, para efectuar reservas debe pertenecer a un proyecto (esta condición es la que permite denominar al usuario "participante"). Dentro de los casos de uso, se determina que el concepto "gestionar" refiere a funciones básicas CRUD dentro de la capa de persistencia del sistema. Por ejemplo, cuando se define que el jefe de proyecto puede "Gestionar proyectos", esto quiere decir que la persona que cumpla este rol puede:

- Consultar y visualizar características de un proyecto que haya inscrito o agregado.
- Editar o actualizar atributos de un proyecto que haya inscrito o agregado.
- Borrar o deshabilitar proyectos que haya inscrito o agregado.

De manera análoga esto se cumple para el bodeguero que puede gestionar insumos, equipamiento y espacios. Adicionalmente, el concepto "evaluar" determina que un rol puede aprobar, rechazar o resolver una instancia de apelación basadas en una reprobación. Por ejemplo, un operador que debe "evaluar reservas", puede ejecutar las siguientes acciones:

- Aprobar una reserva realizada por un participante.
- Rechazar una reserva, pudiendo generar una instancia de apelación de esta decisión.
- Resolver una apelación de rechazo para una reserva.

La figura 18 muestra un resumen gráfico general de los casos de uso abarcados por la plataforma.

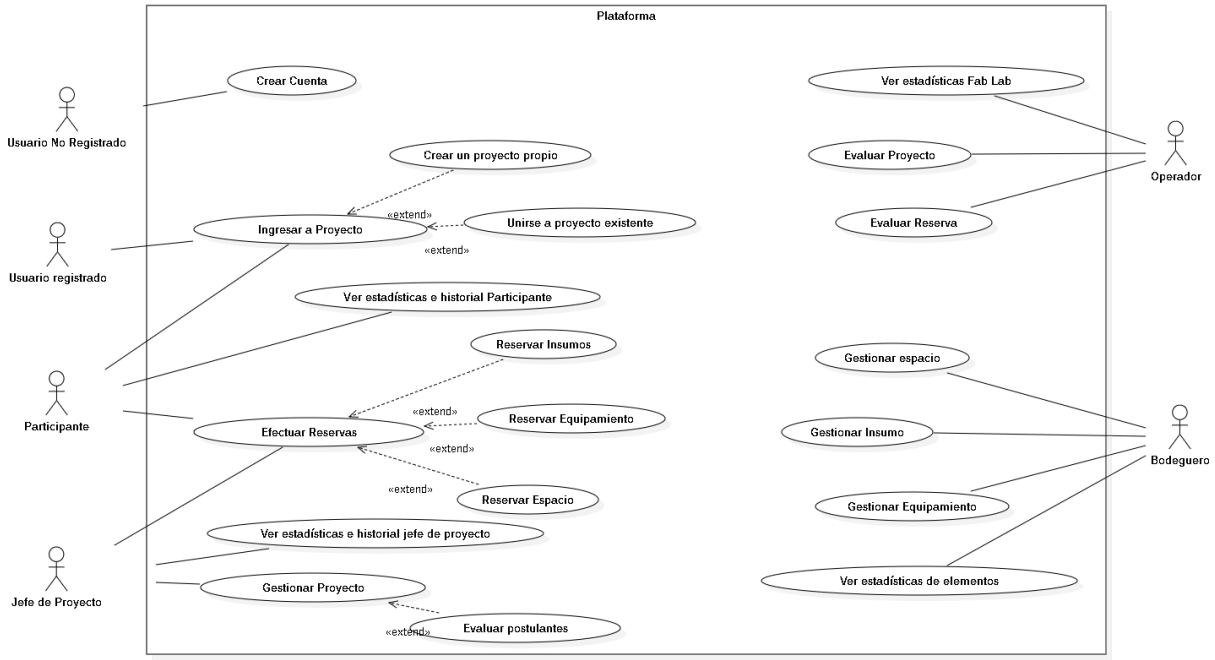


Figura 18: Casos de Uso para la plataforma del Fab Lab.
Fuente: Elaboración Propia

A continuación se presentan los casos de uso narrativos para cada uno.

3.4.1. CREAR CUENTA

- Código: CU-01
- Actor(es): Usuario no registrado
- Precondición(es): No aplica
- Resumen: El sistema va a almacenar los datos de un usuario no registrado que desea utilizar los recursos del Fab Lab para efectuar un proyecto.
- Flujo Normal:
  1. Usuario no registrado solicita registrarse en la plataforma.
  2. El sistema muestra un formulario con datos a ingresar al usuario no registrado.
  3. Usuario no registrado ingresa los datos requeridos por la plataforma en el formulario.
  4. Sistema valida los datos ingresados por usuario no registrado.

5. El sistema informa que el registro se ha realizado correctamente.

- Postcondición(es): Usuario se registró y almacenó en el sistema.
- Flujo Alternativo:

5. El sistema informa que faltan datos por llenar en el formulario al usuario no registrado.
   a) El usuario no registrado llena los datos faltantes en el formulario.
   b) Sistema valida los datos ingresados por usuario no registrado.
   c) El sistema informa que el registro se ha realizado correctamente.

La figura 19 indica los pasos a efectuar para crear una cuenta en la plataforma.

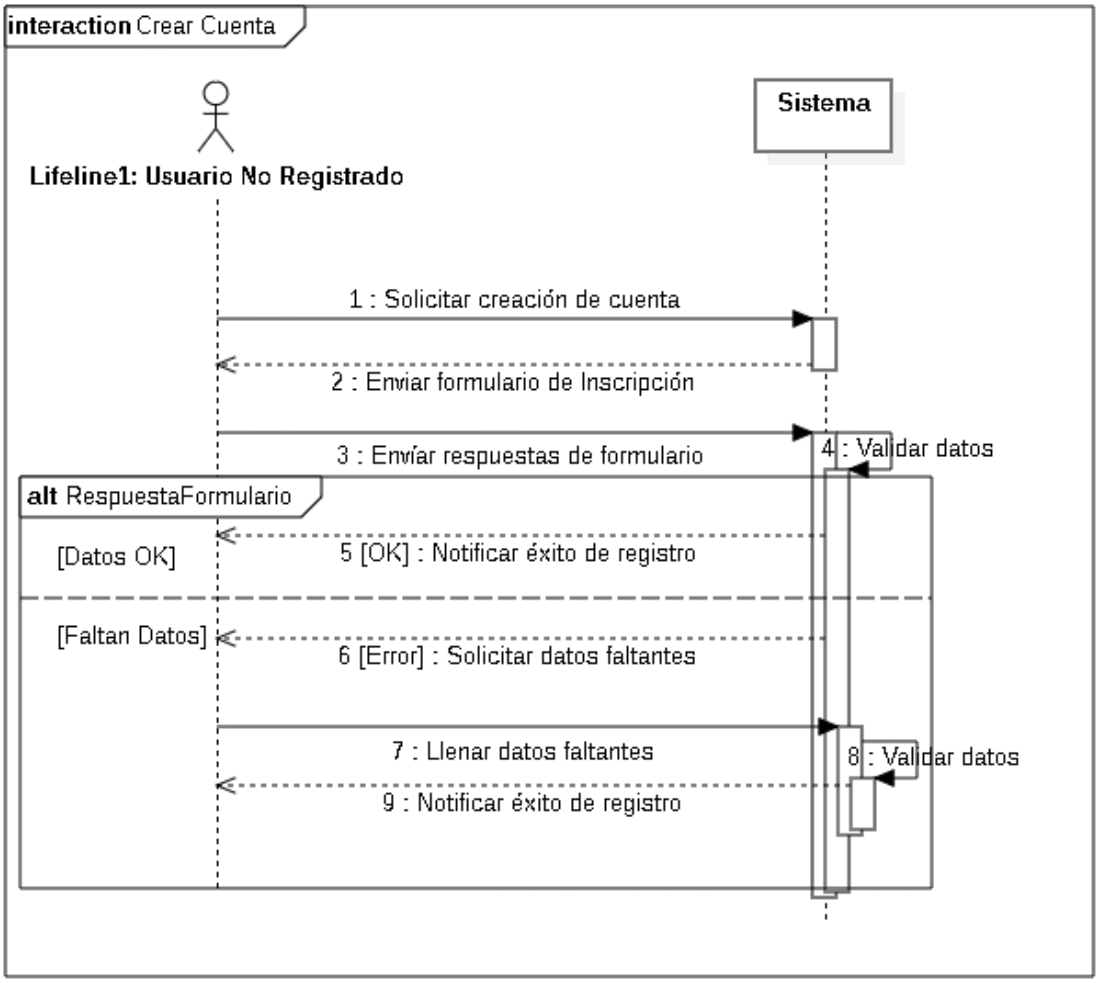


Figura 19: Diagrama de Secuencia para Crear Cuenta.
Fuente: Elaboración Propia

3.4.2. CREAR PROYECTO PROPIO

- Código: CU-02
- Actor(es): Usuario registrado
- Precondición(es): Usuario debe estar registrado en el sistema
- Resumen: El usuario registrado va a solicitar aprobación de su proyecto ingresando los datos necesarios en el sistema de Fab Lab con el propósito de tener derecho a reservar recursos. Sujeto a aprobación de operador del Fab Lab.
- Flujo Normal:
  1. Usuario registrado solicita crear un proyecto en el sistema.
  2. El sistema muestra un formulario con datos a ingresar del proyecto al usuario registrado.
  3. Usuario registrado ingresa los datos requeridos por la plataforma en el formulario.
  4. Sistema valida los datos ingresados por usuario registrado.
  5. El sistema informa que la solicitud se ha realizado correctamente.
- Postcondición(es): Usuario registró solicitud de proyecto exitosamente.
- Flujo Alternativo:
  5. El sistema informa que faltan datos por llenar al usuario registrado.
     a) El usuario registrado llena y envía los datos faltantes en el formulario.
     b) Sistema valida los datos ingresados por usuario registrado.
     c) El sistema informa que la solicitud se ha realizado correctamente.
  5. El sistema informa los datos incorrectos por corregir a usuario registrado.
     a) El usuario registrado llena datos correctamente en el formulario.
     b) Sistema valida los datos ingresados por usuario registrado.
     c) El sistema informa que la solicitud se ha realizado correctamente.

La figura 20 expone en detalle los pasos para crear proyectos.

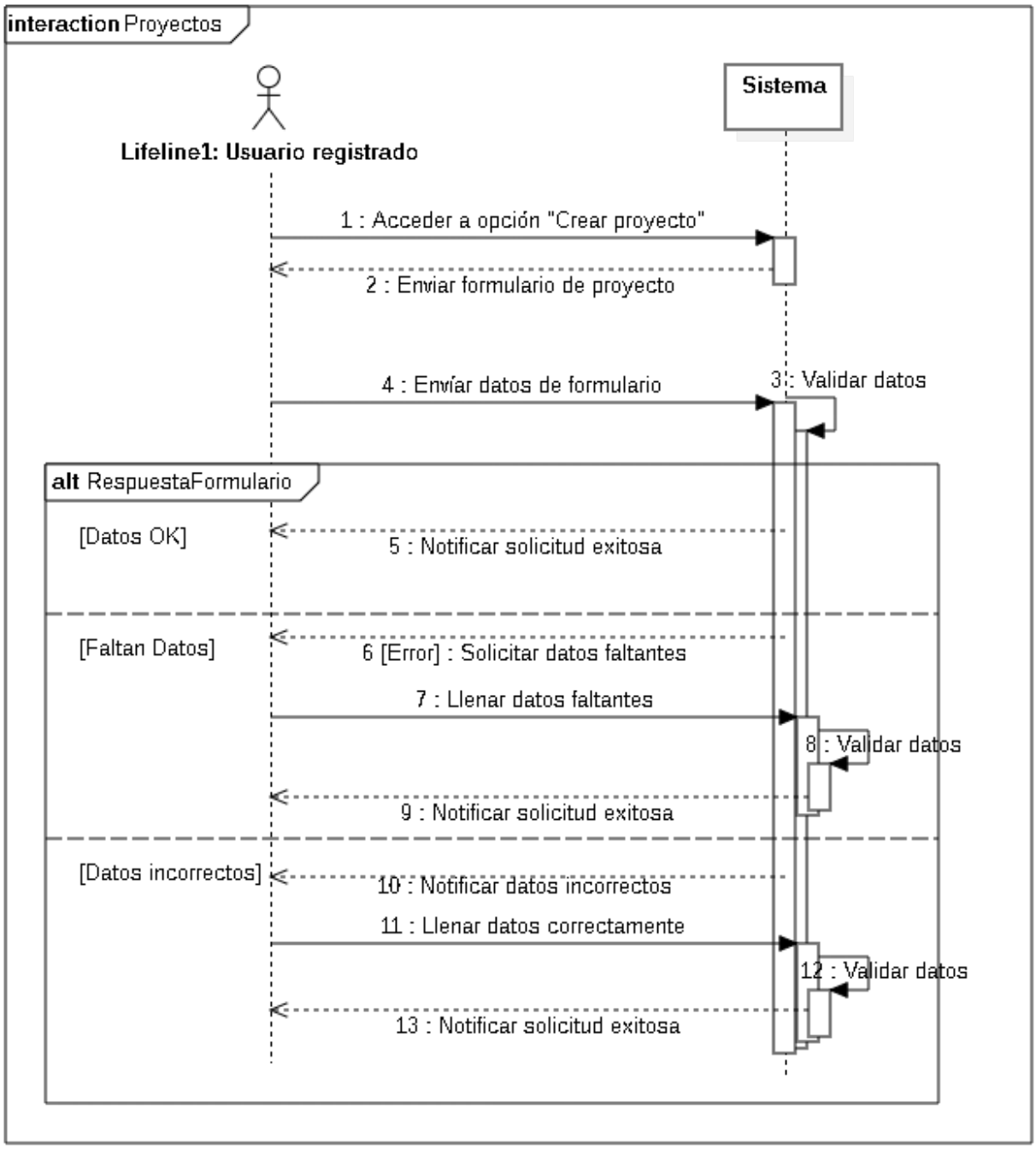


Figura 20: Diagrama de Secuencia en creación de proyectos.
Fuente: Elaboración Propia

3.4.3. EVALUAR PROYECTO

- Código: CU-03
- Actor(es): Operador
- Precondición(es): Proyecto debe estar preinscrito por Jefe de Proyecto.
- Resumen: Operador ve el listado de proyectos postulantes al Fab Lab, por lo que ve sus propiedades y revisa prioridades para aprobarlos.
- Flujo Normal:
  1. Operador selecciona listado de proyectos sin revisar.
  2. El sistema muestra un listado con los proyectos sin revisar.
  3. Operador selecciona un proyecto del listado para ver en detalle.
  4. Sistema muestra detalles del proyecto.
  5. Operador evalúa detalles basados en criterios internos del Fab Lab.
  6. Operador de Fab Lab aprueba proyecto para reservas.
  7. El sistema informa que la evaluación se ha registrado correctamente.
- Postcondición(es): Operador evaluó exitosamente un proyecto en el sistema.
- Flujo Alternativo:
  5. Operador de Fab Lab reprueba proyecto para reservas
     a) El sistema informa que la resolución se ha registrado correctamente.
  5. Operador de Fab Lab responde apelación a rechazo de proyecto.
     a) El sistema informa que la respuesta ha sido exitosamente ingresada.
  4. Sistema muestra apelaciones a rechazo de proyecto.
     a) Operador evalúa detalles basados en criterios internos del Fab Lab.
     b) Operador de Fab Lab responde apelación a rechazo de proyecto.
     c) El sistema informa que la respuesta ha sido exitosamente ingresada.

La evaluación de proyectos estructura sus tareas según la figura 21.

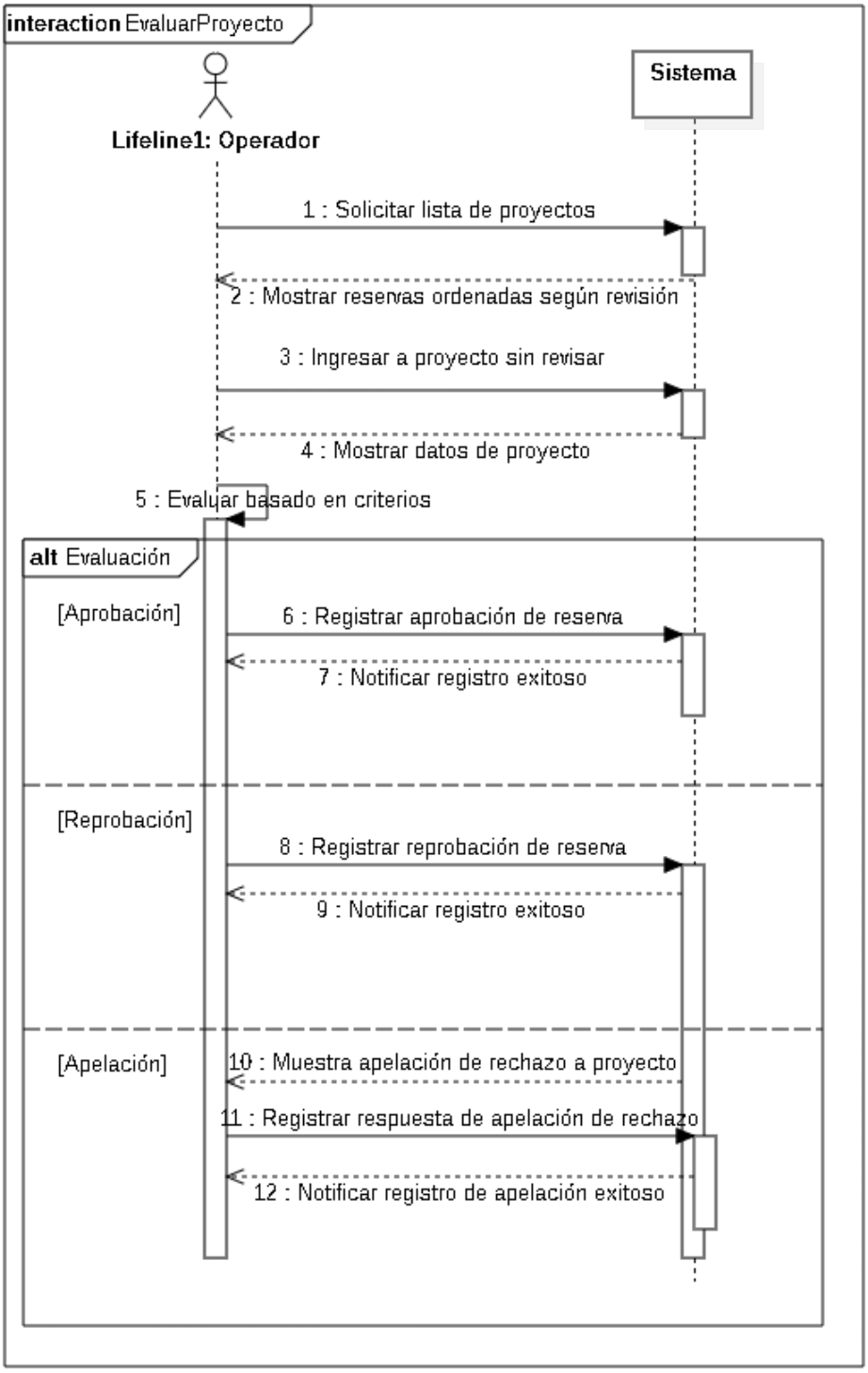


Figura 21: Diagrama de Secuencia para Evaluar Proyectos.
Fuente: Elaboración Propia

3.4.4. GESTIONAR PROYECTO

- Código: CU-04
- Actor(es): Jefe de Proyecto
- Precondición(es): Jefe de proyecto creó un proyecto propio.
- Resumen: Un jefe de proyecto que desea editar las propiedades intrínsecas de sus proyectos en el sistema o deshabilitar su proyecto acorde a lo descrito en la sección 3.4.
- Flujo Normal:
  1. Jefe de proyecto accede a la sección de sus proyectos propios.
  2. El sistema muestra un listado con los proyectos creados por el usuario.
  3. Jefe de proyecto selecciona uno de sus proyectos.
  4. El sistema muestra detalles de proyecto seleccionado por jefe de proyecto. Muestra opciones para editar o deshabilitar.
  5. Jefe de proyecto selecciona editar proyecto.
  6. El sistema muestra un formulario con datos editables a ingresar del proyecto al jefe de proyecto.
  7. Jefe de proyecto edita datos requeridos por la plataforma en el formulario.
  8. El sistema informa que el proyecto se ha modificado correctamente.
- Postcondición(es): Usuario modificó proyecto satisfactoriamente.
- Flujo Alternativo:
  5. Jefe de proyecto selecciona deshabilitar proyecto.
     a) El sistema pide confirmación para deshabilitar proyecto a jefe de proyecto.
     b) Jefe de proyecto confirma deshabilitar proyecto a sistema.
     c) Sistema informa a jefe de proyecto que proyecto fue deshabilitado satisfactoriamente.
  4. Sistema muestra rechazo de solicitud de proyecto.
     a) Jefe de proyecto apela rechazo de proyecto.
     b) Sistema otorga formulario para justificar apelación.
     c) Jefe de proyecto llena formulario y envía a sistema.
     d) Sistema informa a jefe de proyecto que apelación se registra exitosamente.

Para gestionar los proyectos, la secuencia de tareas se grafica en la figura 22.

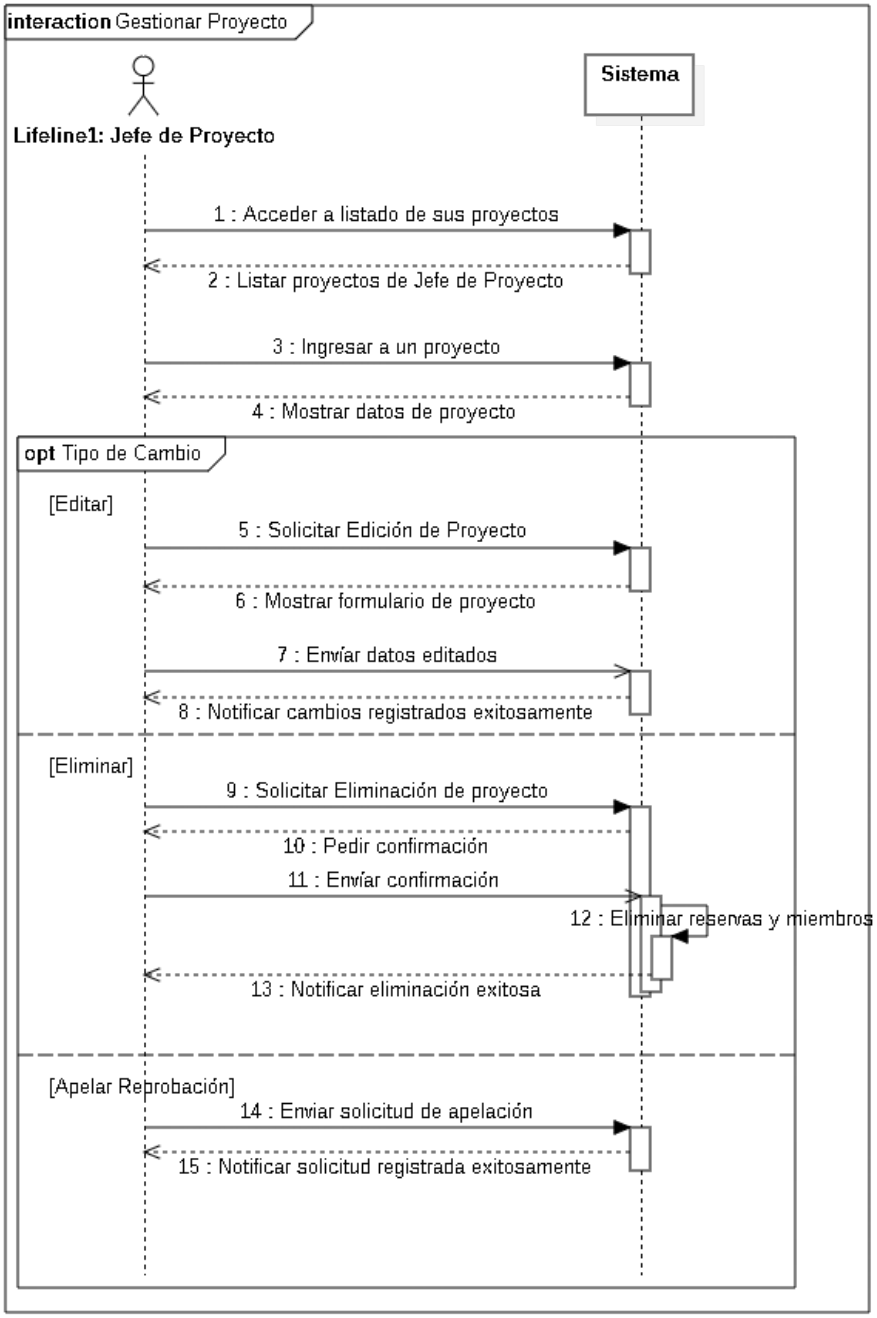


Figura 22: Diagrama de Secuencia para Gestionar Proyecto.
Fuente: Elaboración Propia

3.4.5. UNIRSE A PROYECTO EXISTENTE

- Código: CU-05
- Actor(es): Usuario registrado
- Precondición(es): Usuario debe estar registrado en el sistema.
- Resumen: El usuario registrado busca unirse a un proyecto existente dentro del sistema, dentro de los creados por otros usuarios y que tengan disponibilidad.
- Flujo Normal:
  1. Usuario registrado solicita ver listado de proyectos en el sistema.
  2. El sistema muestra un listado de proyectos.
  3. Usuario registrado ingresa a uno de los proyectos para ver más información.
  4. El sistema muestra información del proyecto al que accede el usuario registrado. Si existe disponibilidad muestra opción para postular a proyecto.
  5. El sistema muestra información del proyecto al que accede el usuario registrado. Si existe disponibilidad muestra opción para postular a proyecto, junto a cuadro de texto para describir motivación e interés en el proyecto.
  6. Usuario registrado llena los datos y postula para unirse a proyecto.
  7. Sistema informa que postulación ha sido registrada satisfactoriamente.
- Postcondición(es): Usuario postuló a proyecto de manera exitosa.
- Flujo Alternativo: No aplica

La figura 23 demuestra el comportamiento de los diferentes actores cuando un usuario registrado desea unirse a un proyecto.

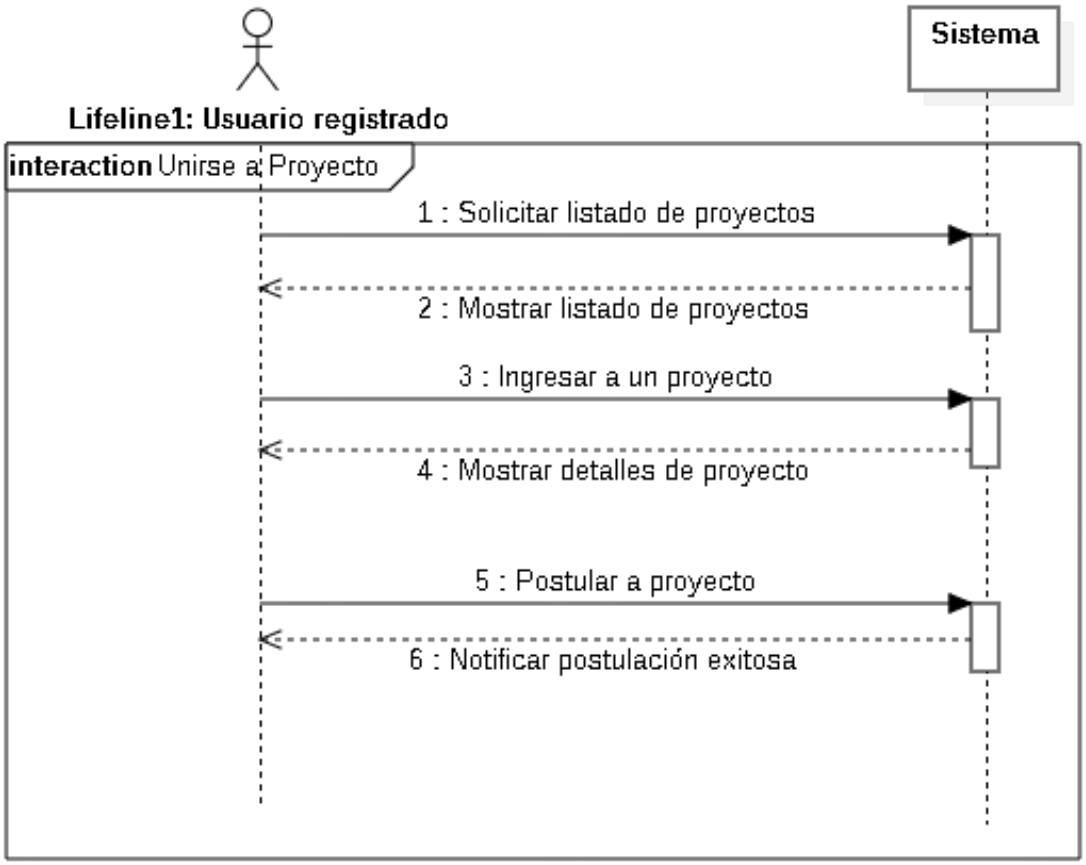


Figura 23: Diagrama de Secuencia para Unirse a Proyecto.
Fuente: Elaboración Propia

3.4.6. EVALUAR POSTULANTE

- Código: CU-06
- Actor(es): Jefe de Proyecto
- Precondición(es): Jefe de proyecto debe tener inscrito al menos un proyecto en la plataforma del Fab Lab.
- Resumen: Jefe de Proyecto ve el listado de postulantes a un proyecto propio, por lo que ve sus motivaciones para aprobarlos.
- Flujo Normal:
    1. Jefe de proyecto accede a listado de proyectos propios.

2. El sistema muestra listado de proyectos propios a jefe de proyecto.
3. Jefe de proyecto selecciona un proyecto del listado para ver en detalle.
4. Sistema muestra detalles del proyecto, sus miembros y postulaciones con evaluación pendiente.
5. Jefe de proyecto selecciona una postulación del listado para ver detalle.
6. Sistema muestra detalles de postulación, y opción para evaluar.
7. Jefe de proyecto evalúa detalles basados en criterios internos del proyecto.
8. Jefe de Proyecto aprueba postulación.
9. El sistema informa que la evaluación se ha registrado correctamente.

- Postcondición(es): Jefe de proyecto aprobó exitosamente un postulante en el sistema.
- Flujo Alternativo:

7. Jefe de Proyecto reprueba postulación
   a) El sistema informa que la resolución se ha registrado correctamente.
4. Sistema muestra apelaciones a reprobación de postulaciones.
   a) Jefe de proyecto evalúa detalles basados en criterios internos del proyecto.
   b) Jefe de proyecto dictamina resolución en plataforma.
   c) El sistema informa que la respuesta ha sido exitosamente ingresada.

La evaluación de postulaciones estructura sus tareas según la figura 24.

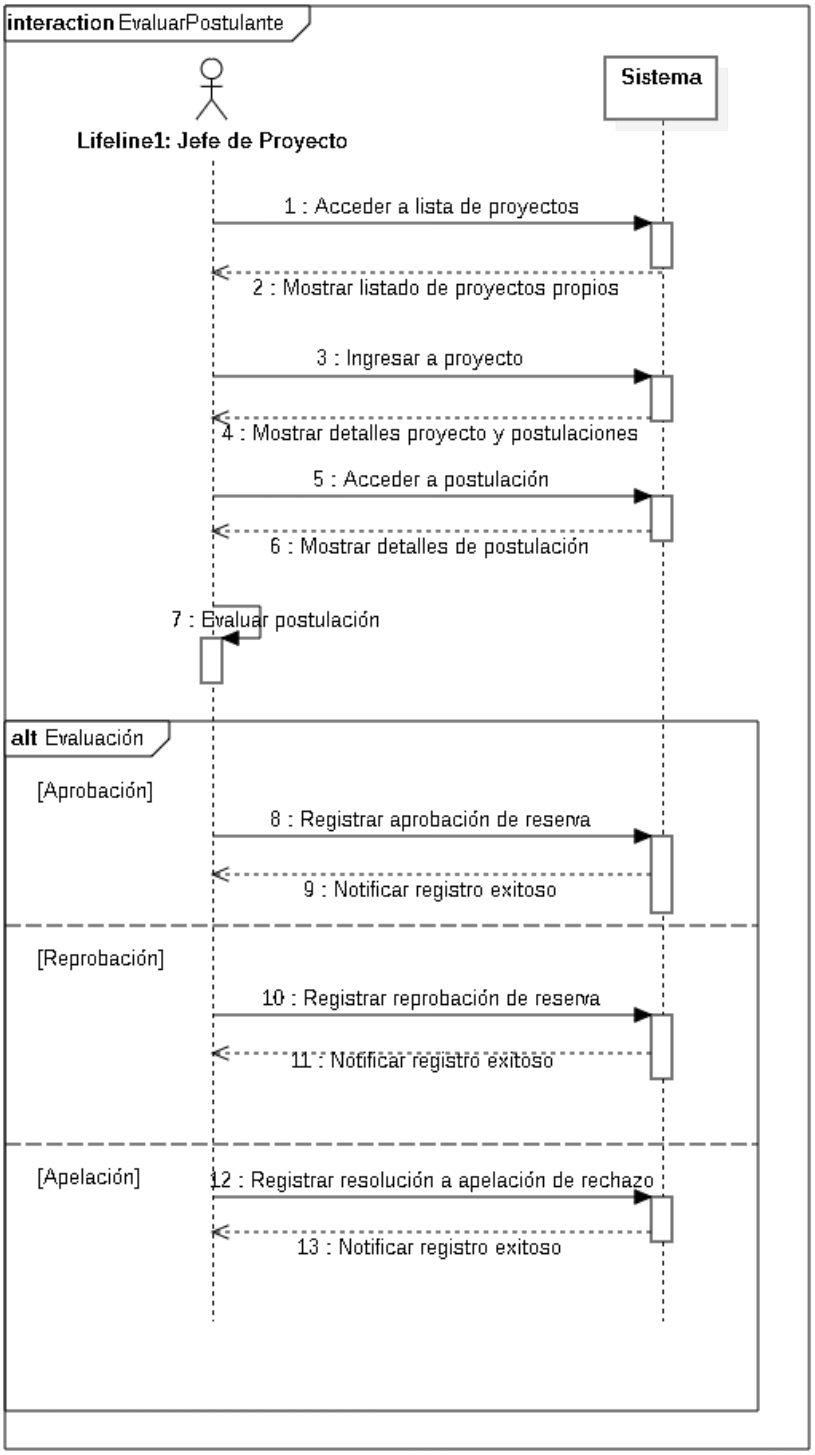


Figura 24: Diagrama de Secuencia para Evaluar postulación.
Fuente: Elaboración Propia

3.4.7. EFECTUAR RESERVA

- Código: CU-07
- Actor(es): Participante
- Precondición(es): Usuario debe estar registrado en el sistema y ser miembro de un proyecto aprobado.
- Resumen: Participante busca efectuar una reserva de algún elemento disponible en el Fab Lab.
- Flujo Normal:
  1. Participante selecciona opción para efectuar reserva en el sistema.
  2. El sistema muestra un listado con los equipos, insumos y espacios disponibles para reservar dentro del Fab Lab a participante.
  3. Participante selecciona elementos que necesita del listado otorgado por sistema.
  4. Sistema muestra fechas y bloques para seleccionar período de reserva a participante.
  5. Participante indica fecha y bloque de inicio, fecha y bloque final al sistema.
  6. Sistema valida los datos ingresados por participante.
  7. El sistema informa que la solicitud de reserva se ha inscrito correctamente.
- Postcondición(es): Solicitud de reserva se registró y almacenó con éxito en el sistema.
- Flujo Alternativo:
  7. Sistema informa elementos ya reservados para fechas y bloques escogidos por participante.
     a) Participante escoge nuevas fechas y bloques para reserva en el sistema.
     b) Sistema valida los datos ingresados por participante.
     c) El sistema informa que la solicitud de reserva se ha inscrito correctamente.

Para cumplir con las reservas, las actividades que se llevan a cabo se encuentran en la figura 25.

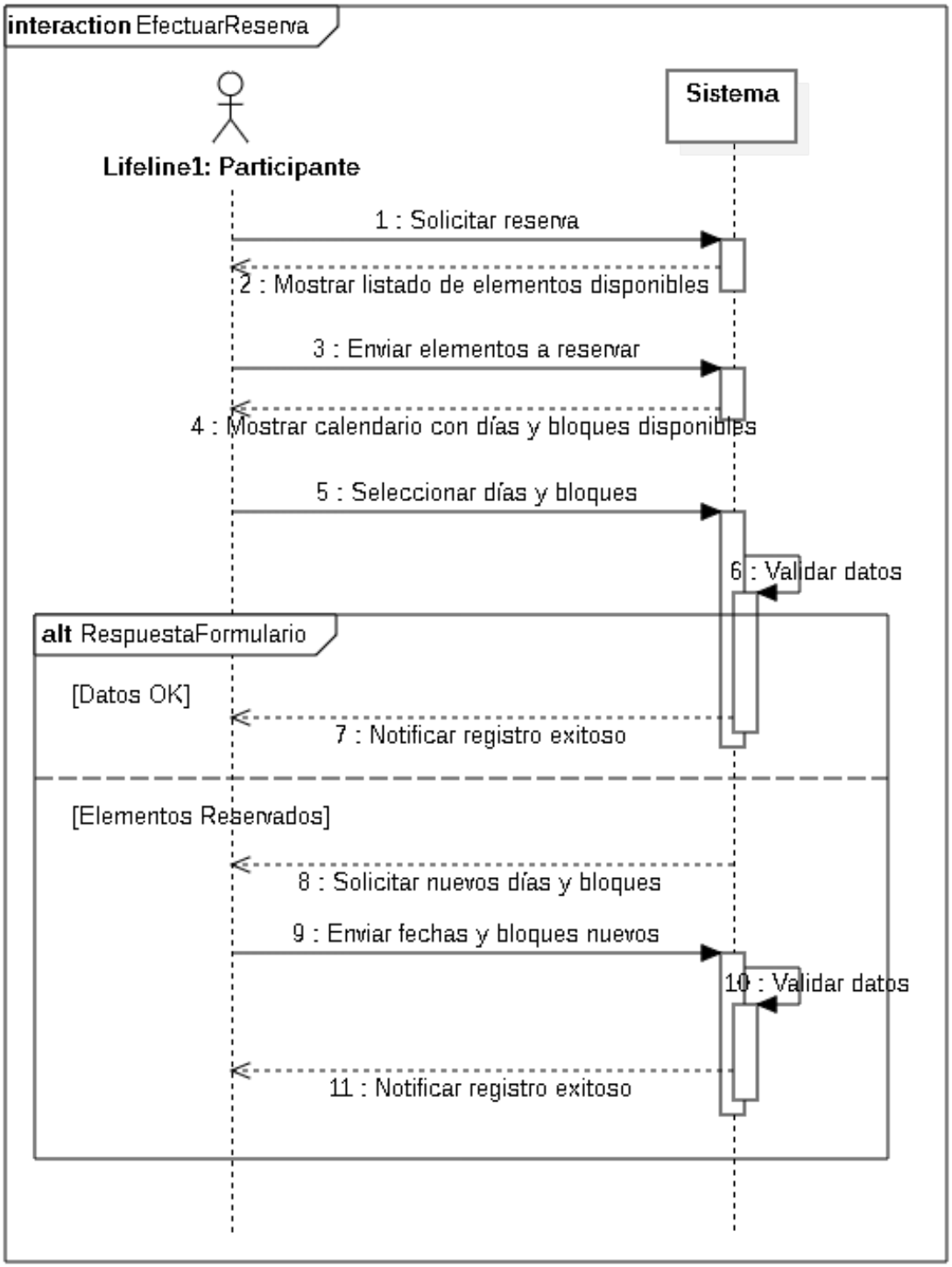


Figura 25: Diagrama de Secuencia para Efectuar Reserva.
Fuente: Elaboración Propia

3.4.8. EVALUAR RESERVA

- Código: CU-08
- Actor(es): Operador
- Precondición(es): Reserva debe estar preinscrita por Participante.
- Resumen: Operador ve el listado de reservas realizadas al Fab Lab por parte de los participantes de algún proyecto, y efectúa filtros de evaluación para aprobarlas.
- Flujo Normal:
  1. Operador selecciona listado de reservas sin revisar.
  2. El sistema muestra un listado con los reservas sin revisar.
  3. Operador selecciona una reserva del listado para ver en detalle.
  4. Sistema muestra detalles de reserva ingresada por participante de proyecto.
  5. Operador evalúa detalles basados en criterios de Fab Lab.
  6. Operador de Fab Lab envía aprobación de reserva para proyecto de participante.
  7. El sistema informa que la aprobación se ha registrado correctamente.
- Postcondición(es): Operador aprobó exitosamente una reserva en el sistema.
- Flujo Alternativo:
  6. Operador rechaza solicitud de reserva
     a) El sistema informa que dictamen se ha registrado correctamente.
  4. Sistema muestra apelación a reprobación de solicitud de reserva.
     a) Operador evalúa detalles basados en criterios de Fab Lab.
     b) Operador dictamina resolución de apelación de participante en la plataforma.
     c) El sistema informa que la resolución ha sido exitosamente ingresada.

La evaluación de reservas ordena sus tareas de maner análoga a la evaluación de proyectos, tal como muestra la figura 26.

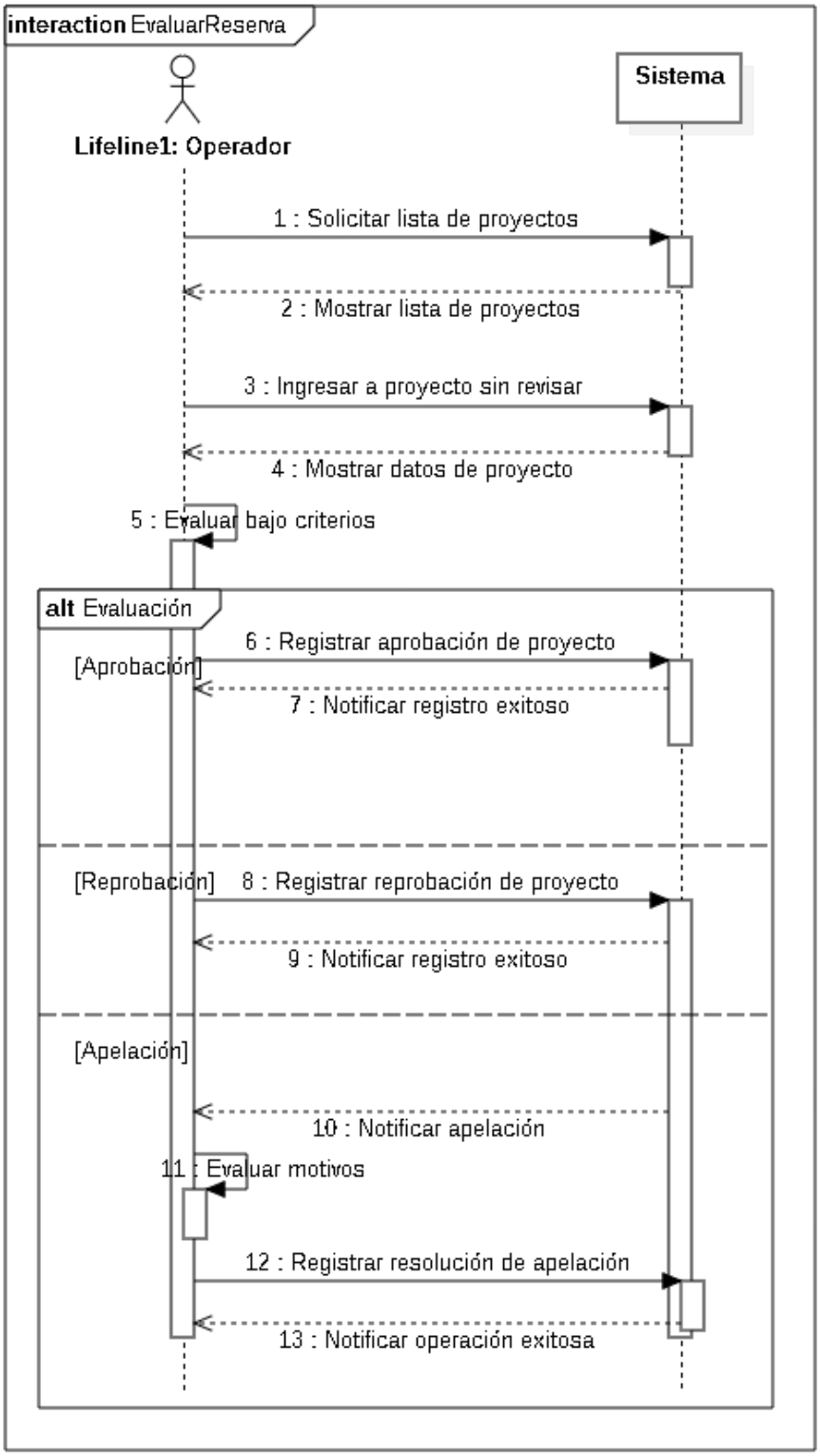


Figura 26: Diagrama de Secuencia para Evaluar Reserva.
Fuente: Elaboración Propia

3.4.9. GESTIONAR EQUIPAMIENTO

- Código: CU-09
- Actor(es): Bodeguero
- Precondición(es): No aplica.
- Resumen: Bodeguero desea almacenar las propiedades intrínsecas de los equipos relevantes en el sistema.
- Flujo Normal:
  1. Bodeguero accede a la sección de equipos.
  2. El sistema muestra un listado con los equipos creados por el bodeguero.
  3. Bodeguero solicita crear un equipo en el sistema.
  4. El sistema muestra un formulario con datos a ingresar del equipo al bodeguero.
  5. Bodeguero ingresa los datos requeridos por la plataforma en el formulario.
  6. Sistema valida los datos ingresados por bodeguero.
  7. El sistema informa que el registro se ha realizado correctamente.
- Postcondición(es): Bodeguero guardó equipo en el sistema satisfactoriamente.
- Flujo Alternativo:
  7. Bodeguero registra parcialmente los datos obligatorios pedidos por el formulario.
     a) El sistema informa los datos que faltan por llenar al bodeguero.
     b) Bodeguero llena los datos faltantes en el formulario.
     c) El sistema informa que el registro se ha realizado correctamente.
  3. Bodeguero desea editar equipo.
     a) Bodeguero selecciona uno de los equipos.
     b) El sistema muestra detalles de equipo seleccionado por bodeguero. Muestra opciones para editar o deshabilitar.
     c) Bodeguero selecciona editar equipo.
     d) El sistema muestra un formulario con datos editables a ingresar del equipo al bodeguero.
     e) Bodeguero edita datos requeridos por la plataforma en el formulario.
     f) El sistema valida los datos ingresado por bodeguero.
     g) El sistema informa que el equipo se ha modificado correctamente.

3. Bodeguero desea deshabilitar equipo.
   a) Bodeguero selecciona uno de los equipos.
   b) El sistema muestra detalles de equipo seleccionado por bodeguero. Muestra opciones para editar o deshabilitar.
   c) Bodeguero selecciona editar equipo.
   d) El sistema pide confirmación para deshabilitar equipo a bodeguero.
   e) Bodeguero confirma deshabilitar equipo a sistema.
   f) Sistema informa a bodeguero que equipo fue deshabilitado satisfactoriamente.

La figura 27 expone las labores ejercidas por los distintos actores para gestionar equipamiento. Este modelo se replica al gestionar insumos y espacios.

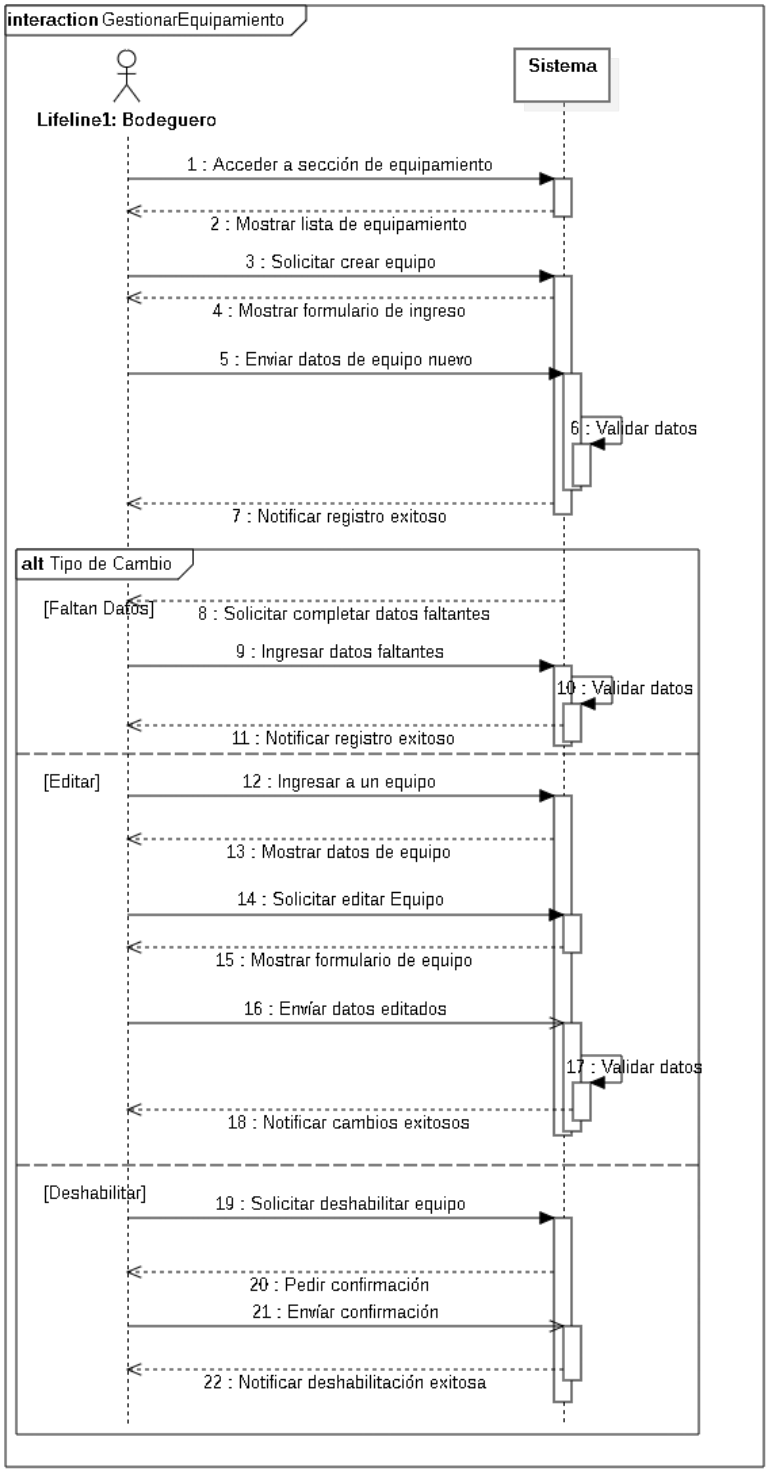


Figura 27: Diagrama de Secuencia para Gestionar Equipamiento.
Fuente: Elaboración Propia

3.4.10. GESTIONAR INSUMO

- Código: CU-10
- Actor(es): Bodeguero
- Precondición(es): No aplica.
- Resumen: Bodeguero desea almacenar las propiedades intrínsecas de los insumos relevantes en el sistema.
- Flujo Normal:
  1. Bodeguero accede a la sección de insumos.
  2. El sistema muestra un listado con los insumos creados por el bodeguero.
  3. Bodeguero solicita crear un insumo en el sistema.
  4. El sistema muestra un formulario con datos a ingresar del insumo al bodeguero.
  5. Bodeguero ingresa los datos requeridos por la plataforma en el formulario.
  6. Sistema valida los datos ingresados por bodeguero.
  7. El sistema informa que el registro se ha realizado correctamente.
- Postcondición(es): Bodeguero guardó insumo en el sistema satisfactoriamente.
- Flujo Alternativo:
  7. Bodeguero registra parcialmente los datos obligatorios pedidos por el formulario.
     a) El sistema informa los datos que faltan por llenar al bodeguero.
     b) Bodeguero llena los datos faltantes en el formulario.
     c) El sistema informa que el registro se ha realizado correctamente.
  3. Bodeguero desea editar insumo.
     a) Bodeguero selecciona uno de los insumos.
     b) El sistema muestra detalles de insumo seleccionado por bodeguero. Muestra opciones para editar o deshabilitar.
     c) Bodeguero selecciona editar insumo.
     d) El sistema muestra un formulario con datos editables a ingresar del insumo al bodeguero.
     e) Bodeguero edita datos requeridos por la plataforma en el formulario.
     f) El sistema valida los datos ingresado por bodeguero.
     g) El sistema informa que el insumo se ha modificado correctamente.

3. Bodeguero desea deshabilitar insumo.
   a) Bodeguero selecciona uno de los insumos.
   b) El sistema muestra detalles de insumo seleccionado por bodeguero. Muestra opciones para editar o deshabilitar.
   c) Bodeguero selecciona editar insumo.
   d) El sistema pide confirmación para deshabilitar insumo a bodeguero.
   e) Bodeguero confirma deshabilitar insumo a sistema.
   f) Sistema informa a bodeguero que insumo fue deshabilitado satisfactoriamente.

La figura 28 detalla las tareas a ejecutar por para gestionar insumo, el cual está basado en la figura 27.

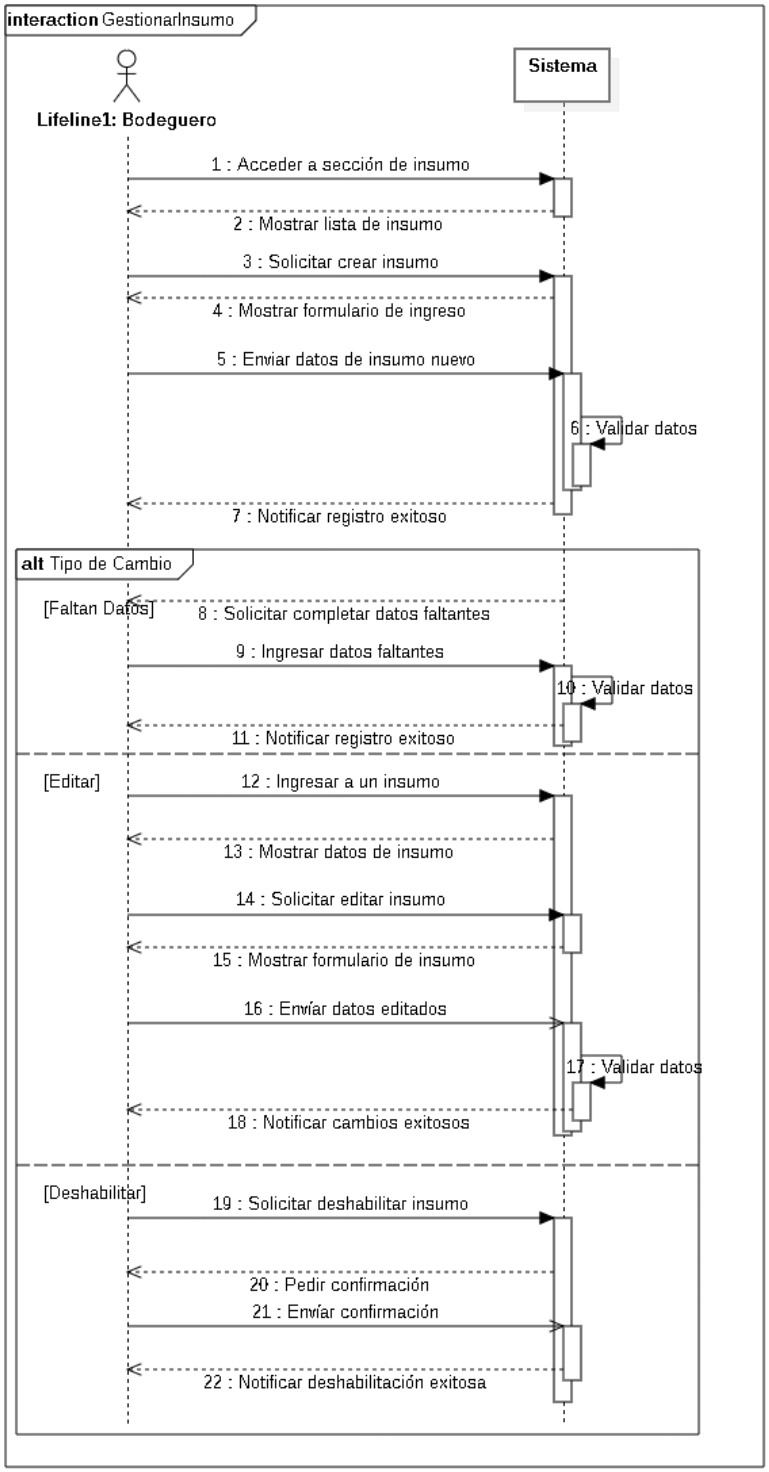


Figura 28: Diagrama de Secuencia para Gestionar Insumo.
Fuente: Elaboración Propia

3.4.11. GESTIONAR ESPACIO

- Código: CU-11
- Actor(es): Bodeguero
- Precondición(es): No aplica.
- Resumen: Bodeguero desea almacenar las propiedades intrínsecas de los espacios disponibles para utilizar en el sistema.
- Flujo Normal:
  1. Bodeguero accede a la sección de espacios.
  2. El sistema muestra un listado con los espacios creados por el bodeguero.
  3. Bodeguero solicita crear un espacio en el sistema.
  4. El sistema muestra un formulario con datos a ingresar del espacio al bodeguero.
  5. Bodeguero ingresa los datos requeridos por la plataforma en el formulario.
  6. Sistema valida los datos ingresados por bodeguero.
  7. El sistema informa que el registro se ha realizado correctamente.
- Postcondición(es): Bodeguero guardó espacio en el sistema satisfactoriamente.
- Flujo Alternativo:
  7. Bodeguero registra parcialmente los datos obligatorios pedidos por el formulario.
     a) El sistema informa los datos que faltan por llenar al bodeguero.
     b) Bodeguero llena los datos faltantes en el formulario.
     c) El sistema informa que el registro se ha realizado correctamente.

  3. Bodeguero desea editar espacio.
     a) Bodeguero selecciona uno de los espacios.
     b) El sistema muestra detalles de espacio seleccionado por bodeguero. Muestra opciones para editar o deshabilitar.
     c) Bodeguero selecciona editar espacio.
     d) El sistema muestra un formulario con datos editables a ingresar del espacio al bodeguero.
     e) Bodeguero edita datos requeridos por la plataforma en el formulario.
     f) El sistema valida los datos ingresado por bodeguero.
     g) El sistema informa que el insumo se ha modificado correctamente.

3. Bodeguero desea deshabilitar espacio.
   a) Bodeguero selecciona uno de los espacios.
   b) El sistema muestra detalles de espacio seleccionado por bodeguero. Muestra opciones para editar o deshabilitar.
   c) Bodeguero selecciona editar espacio.
   d) El sistema pide confirmación para deshabilitar espacio a bodeguero.
   e) Bodeguero confirma deshabilitar espacio a sistema.
   f) Sistema informa a bodeguero que espacio fue deshabilitado satisfactoriamente.

La figura 29 detalla las tareas a ejecutar por para gestionar espacio basado en la figura 27.

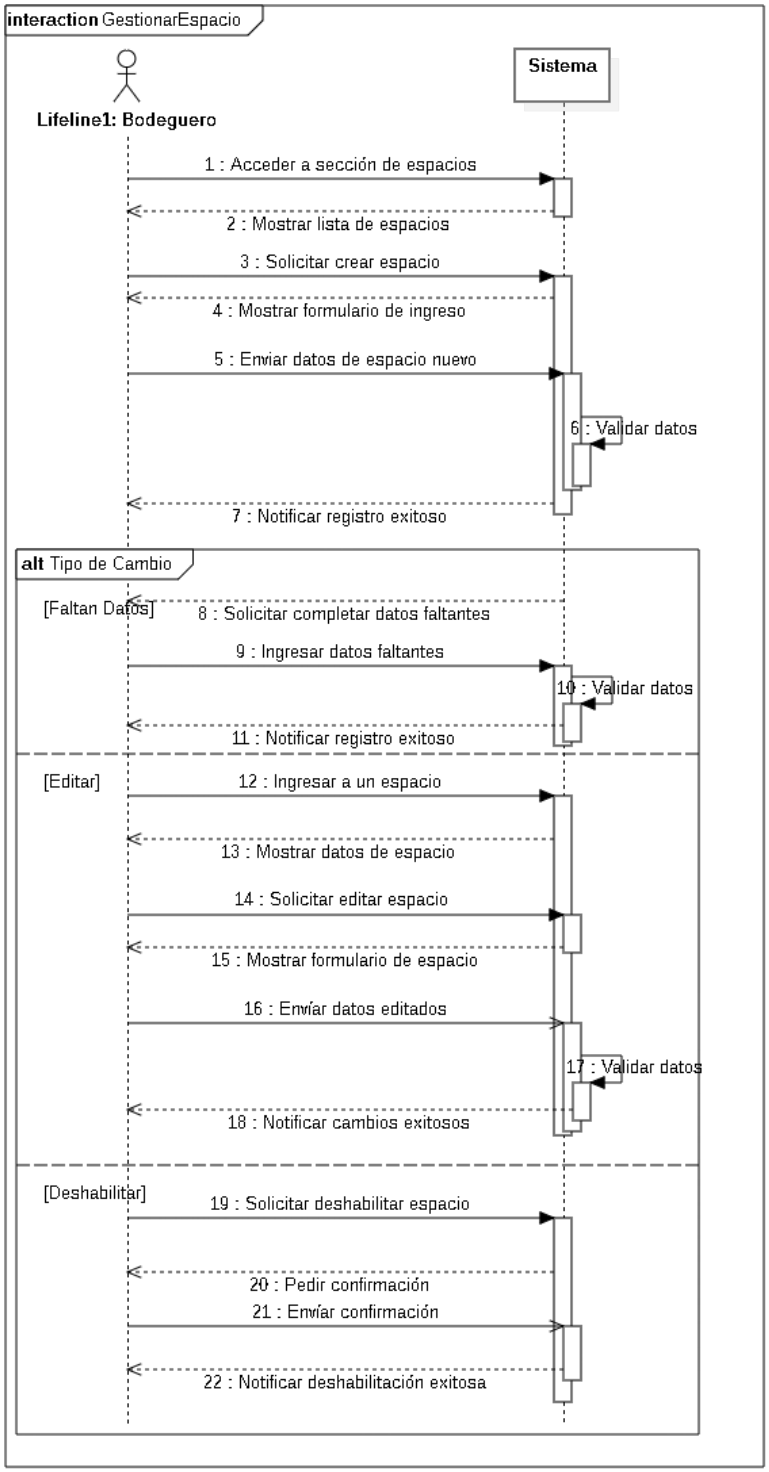


Figura 29: Diagrama de Secuencia para Gestionar Espacio.
Fuente: Elaboración Propia

3.4.12. VER ESTADÍSTICAS E HISTORIAL PARTICIPANTE

- Código: CU-12
- Actor(es): Participante
- Precondición(es): Usuario debe estar registrado en el sistema y ser miembro de un proyecto aprobado.
- Resumen: Participante solicita al sistema mostrar historial y datos estadísticos respecto a su actividad dentro de la plataforma. Las actividades consideradas están relacionadas con los proyectos a los que ha pertenecido y las reservas realizadas en cada uno de ellos.
- Flujo Normal:
  1. Participante accede a opción "Ver estadísticas".
  2. El sistema muestra formulario a participante con los siguientes datos: Fecha de inicio, fecha de término. Adicionalmente, muestra casillas de verificación para los siguientes datos a mostrar: Cantidad total de proyectos a los que pertenece, historial de proyectos a los que ha pertenecido y cargos, cantidad de reservas totales, historial de reservas y proyecto al cual se realiza.
  3. Participante llena el formulario y lo envía a sistema.
  4. Sistema valida los datos ingresados por participante.
  5. El sistema muestra datos históricos y estadísticos seleccionados con posibilidad de filtrar según proyecto, cargo, estado o modificación de fechas de inicio o término a Participante.
- Postcondición(es): Sistema mostró a participante historial y datos estadísticos seleccionados.
- Flujo Alternativo:
  3. Participante ingresa fechas inválidas.
     a) Sistema solicita corregir fechas a participante.
     b) Participante llena el formulario con fechas correctas y lo envía a sistema.
     c) Sistema valida los datos ingresados por participante.
     d) El sistema muestra datos históricos y estadísticos seleccionados con posibilidad de filtrar según proyecto, cargo, estado o modificación de fechas de inicio o término a participante.
  3. Participante no marca ningúna casilla de verificación.
     a) Sistema solicita marcar por lo menos una casilla a participante.
     b) Participante llena el formulario y lo envía a sistema.

c) Sistema valida los datos ingresados por participante.

d) El sistema muestra datos históricos y estadísticos seleccionados con posibilidad de filtrar según proyecto, cargo, estado o modificación de fechas de inicio o término a participante.

La figura 30 indica los pasos a efectuar para que un participante vea estadísticas de historial de su actividad en la plataforma.

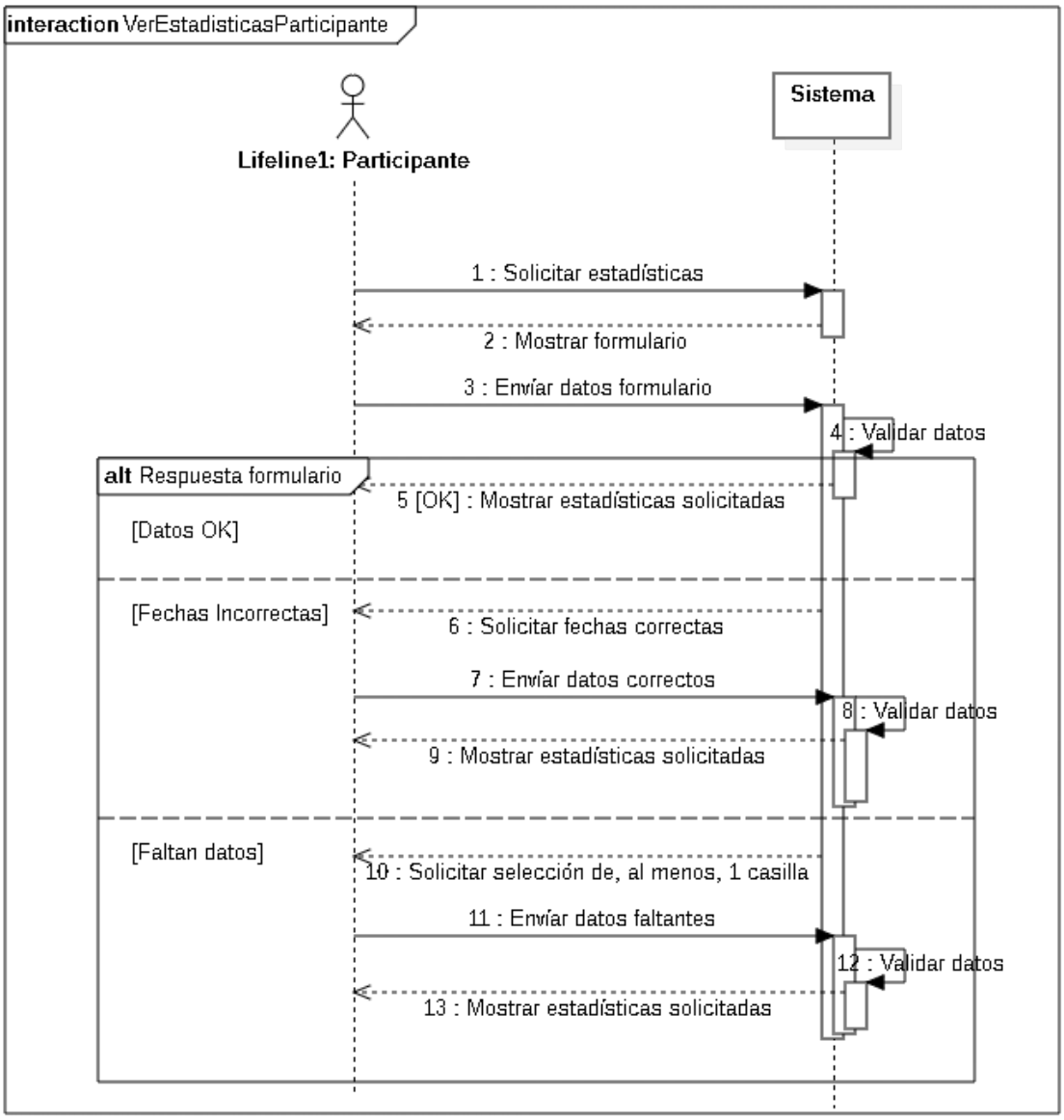


Figura 30: Diagrama de Secuencia para ver estadísticas de participante.
Fuente: Elaboración Propia

3.4.13. VER ESTADÍSTICAS E HISTORIAL JEFE DE PROYECTO

- Código: CU-13
- Actor(es): Jefe de proyecto
- Precondición(es): Usuario debe haber creado un proyecto, proyecto debe haber sido aprobado.
- Resumen: El sistema mostrará historial de jefe de proyecto y datos estadísticos respecto a su actividad dentro de la plataforma.
- Flujo Normal:
  1. Jefe de Proyecto accede a opción "Ver estadísticas".
  2. El sistema muestra formulario a jefe de proyecto con los siguientes datos: Fecha de inicio, fecha de término. Adicionalmente, muestra casillas de verificación para los siguientes datos a mostrar: cantidad total de proyectos a los que pertenece, cantidad de proyectos que ha creado, historial de proyectos y cargos en proyectos propios y en projectos de otros jefes de proyecto, cantidad de reservas totales, historial de reservas y proyecto al cual se realiza.
  3. Jefe de Proyecto llena el formulario y lo envía a sistema.
  4. Sistema valida los datos ingresados por jefe de proyecto.
  5. El sistema muestra datos históricos y estadísticos seleccionados con posibilidad de filtrar según proyecto, cargo, estado o modificación de fechas de inicio o término.
- Postcondición(es): Sistema mostró a Jefe de Proyecto historial y datos estadísticos seleccionados.
- Flujo Alternativo:
  3. Jefe de proyecto ingresa fechas inválidas.
     a) Sistema solicita corregir fechas a Jefe de Proyecto.
     b) Jefe de Proyecto llena el formulario con fechas correctas y lo envía a sistema.
     c) Sistema valida los datos ingresados por participante.
     d) El sistema muestra datos históricos y estadísticos seleccionados con posibilidad de filtrar según proyecto, cargo, estado o modificación de fechas de inicio o término.

  3 Jefe de Proyecto no marca ningúna casilla de verificación.
     a) Sistema solicita marcar por lo menos una casilla a jefe de proyecto.
     b) Jefe de Proyecto llena el formulario y lo envía a sistema.

c) Sistema valida los datos ingresados por participante.

d) El sistema muestra datos históricos y estadísticos seleccionados con posibilidad de filtrar según proyecto, cargo o estado.

La figura 31 indica los pasos a efectuar para que un jefe de proyecto vea estadísticas de historial de su actividad en la plataforma.

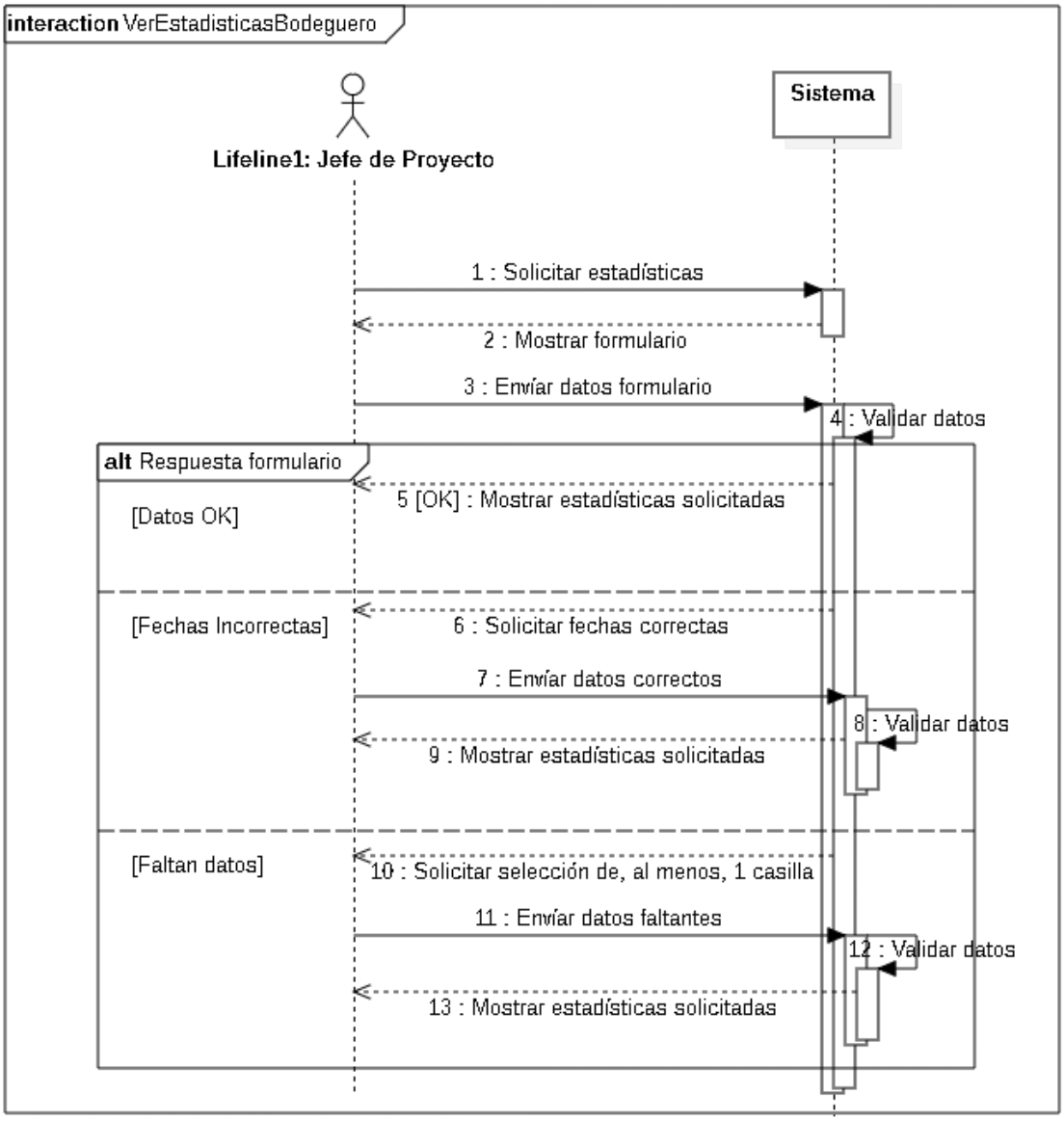


Figura 31: Diagrama de Secuencia para ver estadísticas de Jefe de Proyecto.
Fuente: Elaboración Propia

3.4.14. VER ESTADÍSTICAS DE ELEMENTOS

- Código: CU-14
- Actor(es): Bodeguero
- Precondición(es): Sistema debe tener elementos registrados en el sistema.
- Resumen: El sistema mostrará historial de adquisiciones, cambios de estado de elementos y datos estadísticos dentro de la plataforma a cargo de bodeguero.
- Flujo Normal:
  1. Bodeguero accede a opción "Ver estadísticas".
  2. El sistema muestra formulario a bodeguero con los siguientes datos: Fecha de inicio, fecha de término. Adicionalmente, muestra casillas de verificación para los siguientes datos a mostrar: cantidad total de equipamiento, cantidad de insumos, cantidad de espacios, historial de equipamiento y estados, historial de insumos y estados, historial de espacios y estados.
  3. Bodeguero llena el formulario y lo envía a sistema.
  4. Sistema valida los datos ingresados por bodeguero.
  5. El sistema muestra datos históricos y estadísticos seleccionados con posibilidad de filtrar según equipamiento, espacio, insumo, estado o modificación de fechas de inicio o término.
- Postcondición(es): Sistema mostró a bodeguero historial y datos estadísticos de los elementos seleccionados.
- Flujo Alternativo:
  3. Bodeguero ingresa fechas inválidas.
     a) Sistema solicita corregir fechas a Bodeguero.
     b) Bodeguero llena el formulario con fechas correctas y lo envía a sistema.
     c) Sistema valida los datos ingresados por bodeguero.
     d) El sistema muestra datos históricos y estadísticos seleccionados con posibilidad de filtrar según equipamiento, espacio, insumo, estado o modificación de fechas de inicio o término.
  3. Bodeguero no marca ningúna casilla de verificación.
     a) Sistema solicita marcar por lo menos una casilla a Bodeguero.
     b) Bodeguero llena el formulario y lo envía a sistema.
     c) Sistema valida los datos ingresados por bodeguero.
     d) El sistema muestra datos históricos y estadísticos seleccionados con posibilidad de filtrar según equipamiento, espacio, insumo, estado o modificación de fechas de inicio o término.

La figura 32 indica los pasos a efectuar para que bodeguero vea estadísticas de historial de su actividad en la plataforma.

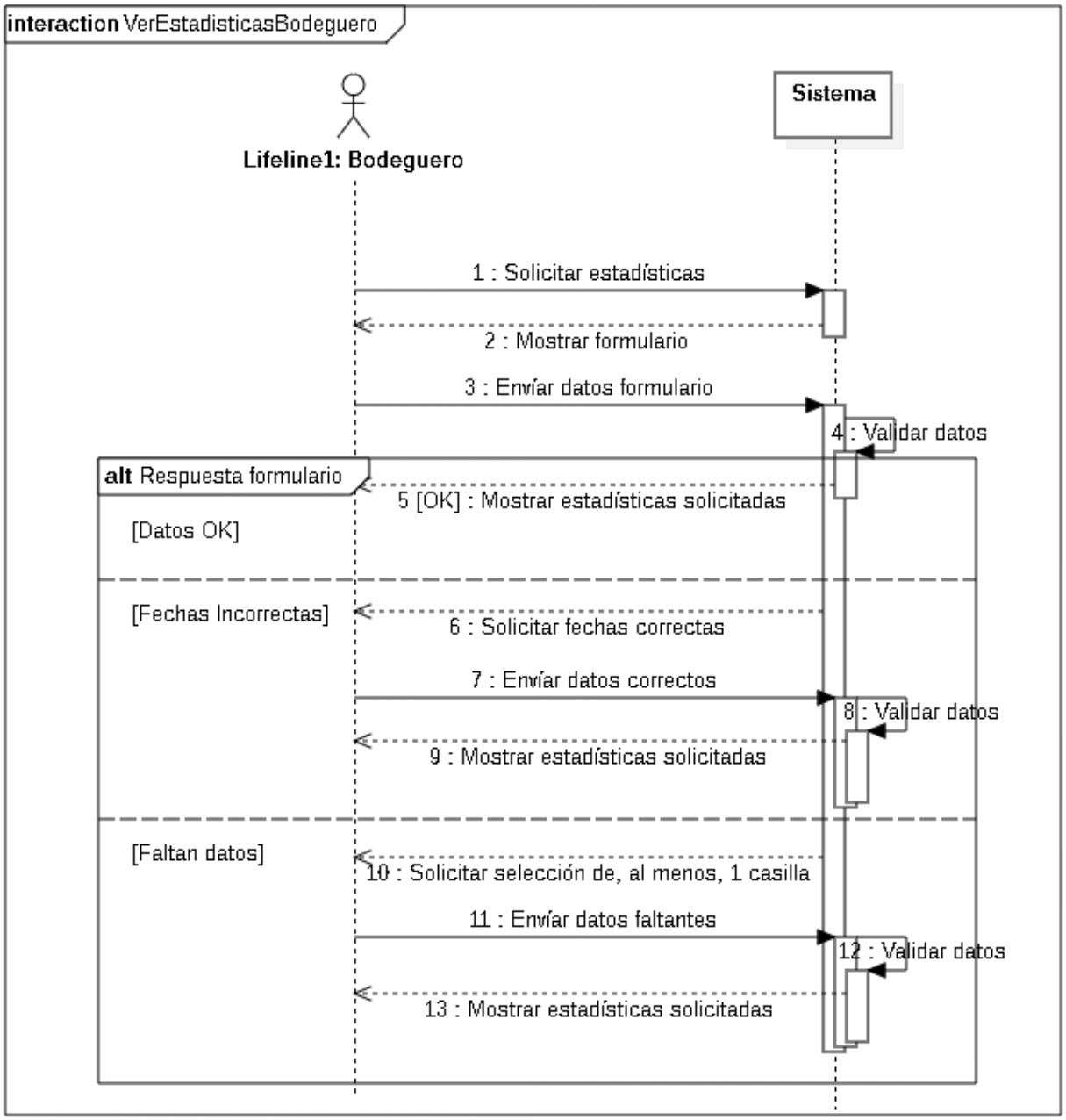


Figura 32: Diagrama de Secuencia para ver estadísticas de Bodeguero.
Fuente: Elaboración Propia

3.4.15. VER ESTADÍSTICAS FAB LAB

- Código: CU-15
- Actor(es): Operador
- Precondición(es): No aplica.
- Resumen: El sistema mostrará datos estadísticos e historial de reservas y proyectos a operador en Fab Lab.
- Flujo Normal:
  1. Operador accede a opción "Ver estadísticas".
  2. El sistema muestra formulario a operador con los siguientes datos: Fecha de inicio, fecha de término. Adicionalmente, muestra casillas de verificación para los siguientes datos a mostrar: cantidad de proyectos con financiamiento, cantidad de proyectos open source, cantidad de proyectos relacionados con USM, cantidad de proyectos totales, cantidad de proyectos aprobados, tasa de proyectos cancelados por unidad de tiempo a elección, historial de proyectos, tasa de reservas por unidad de tiempo a elección, historial de reservas, tasa de uso de equipamiento, insumos y espacios, cantidad de elementos sin disponibilidad.
  3. Operador llena el formulario y lo envía a sistema.
  4. Sistema valida los datos ingresados por operador.
  5. El sistema muestra datos históricos y estadísticos seleccionados con posibilidad de filtrar según proyecto, tipo de proyecto, criterios de aceptación, estado de proyectos, estado de reservas o modificación de fechas de inicio o término.
- Postcondición(es): Sistema mostró a operador historial y datos estadísticos de los elementos seleccionados.
- Flujo Alternativo:
  3. Operador ingresa fechas inválidas.
     a) Sistema solicita corregir fechas a Operador.
     b) Bodeguero llena el formulario con fechas correctas y lo envía a sistema.
     c) Sistema valida los datos ingresados por operador.
     d) El sistema muestra datos históricos y estadísticos seleccionados con posibilidad de filtrar según proyecto, tipo de proyecto, criterios de aceptación, estado de proyectos, estado de reservas o modificación de fechas de inicio o término.
  3. Operador no marca ningúna casilla de verificación.

a) Sistema solicita marcar por lo menos una casilla a operador.
b) Operador llena el formulario y lo envía a sistema.
c) Sistema valida los datos ingresados por operador.
d) El sistema muestra datos históricos y estadísticos seleccionados con posibilidad de filtrar según proyecto, tipo de proyecto, criterios de aceptación, estado de proyectos, estado de reservas o modificación de fechas de inicio o término.

La figura 33 indica los pasos a efectuar para que Operador vea estadísticas de historial de su actividad en la plataforma.

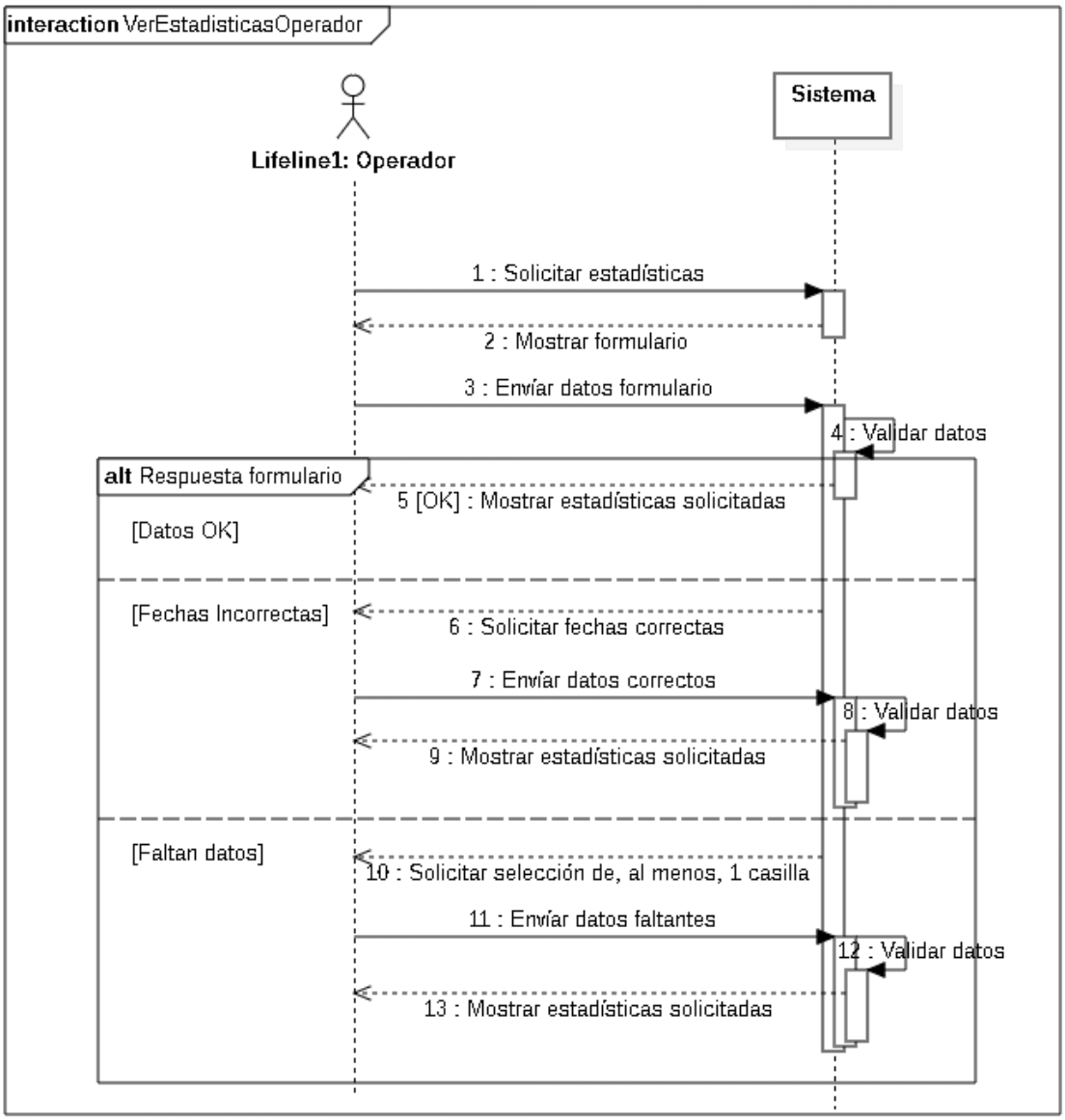


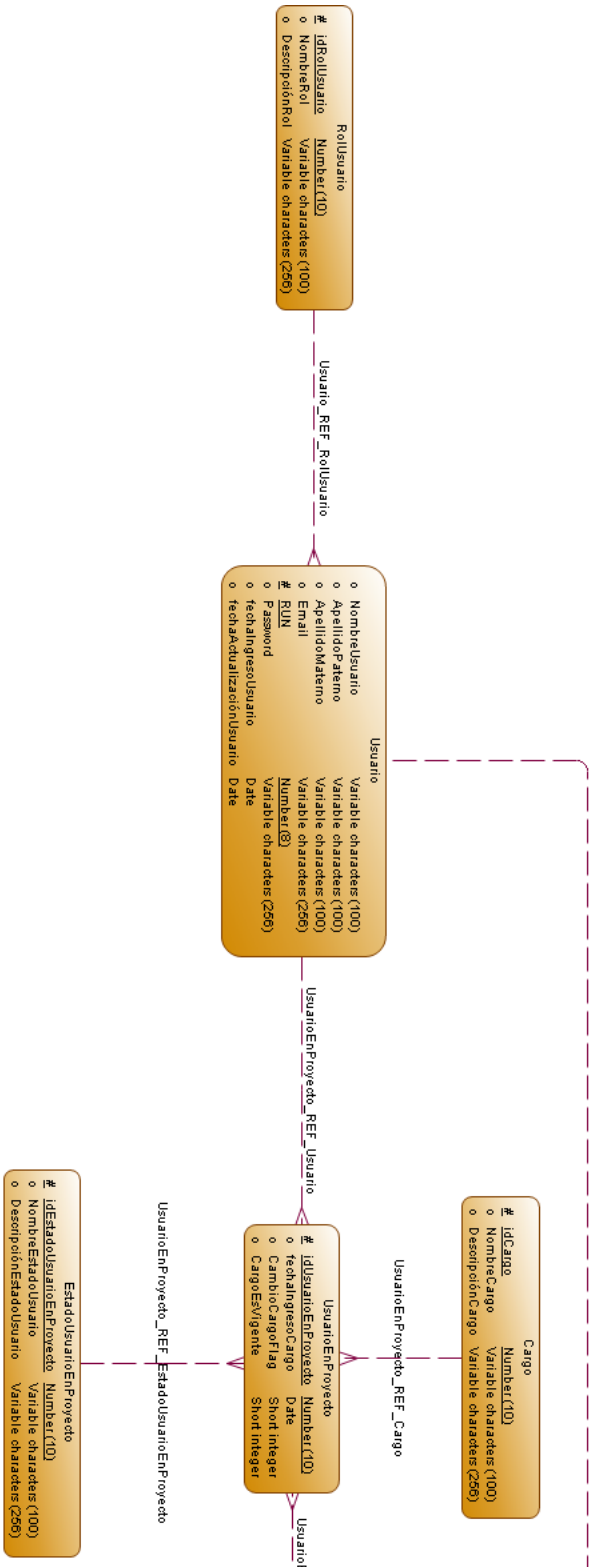
Figura 33: Diagrama de Secuencia para ver estadísticas de Operador.
Fuente: Elaboración Propia

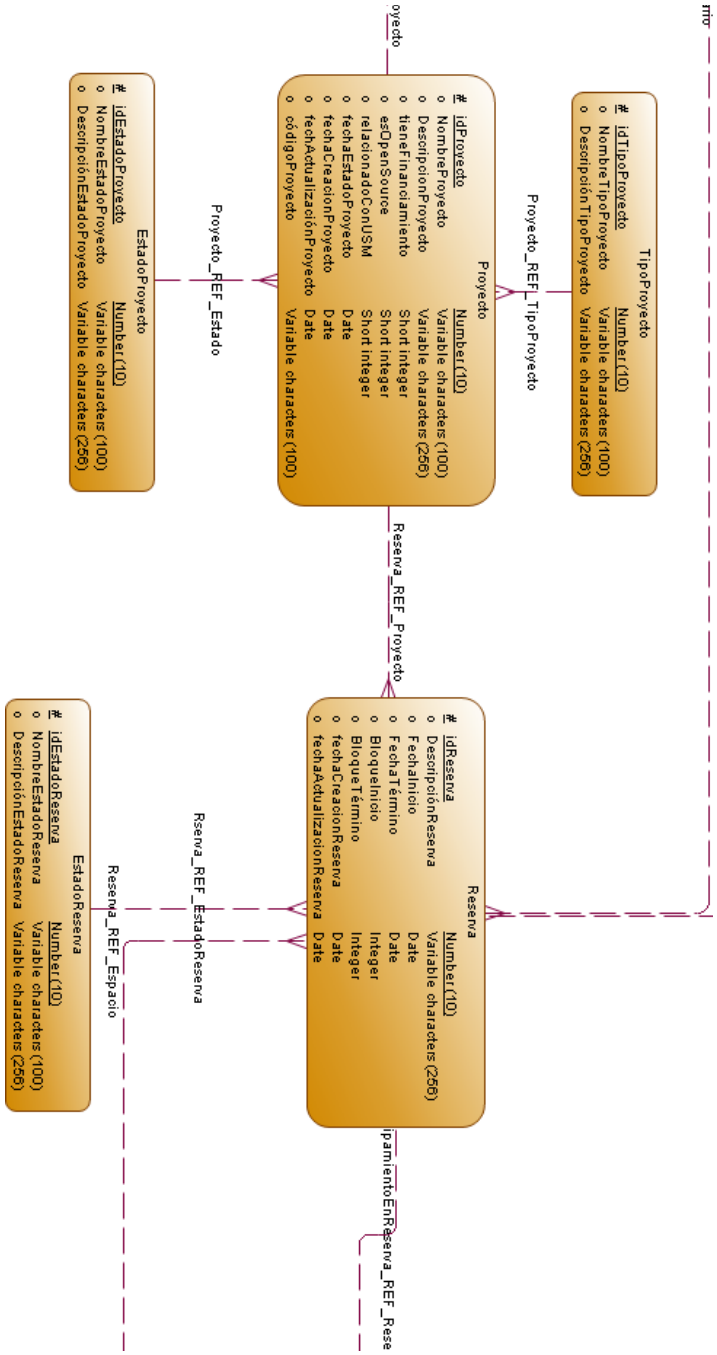
CAPÍTULO 4
DISEÑO DEL SISTEMA

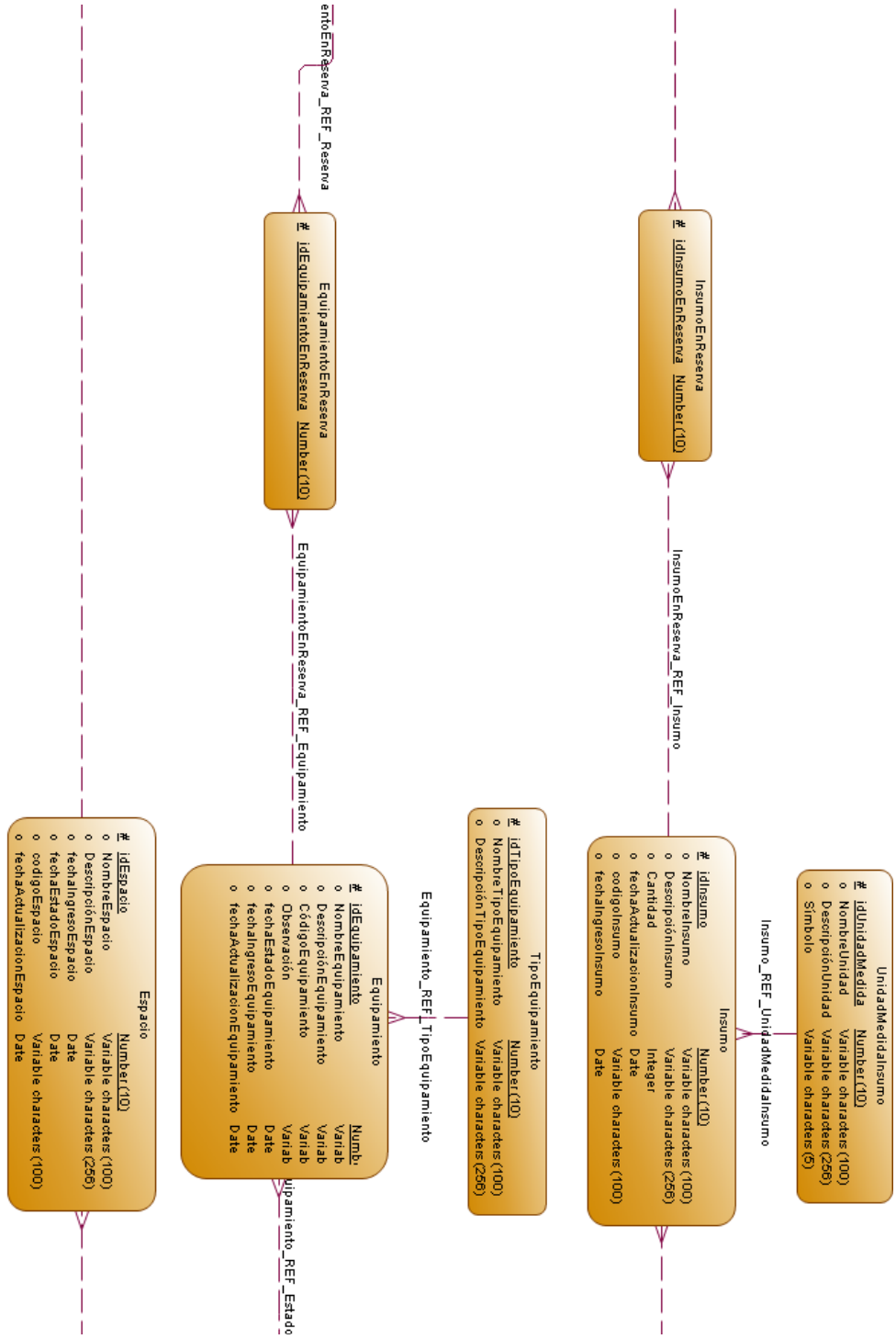
Esta sección explica en detalle cómo se establecen las clases y entidades que definirán la organización de los datos en el sistema, los cuales conforman la estructura base para cumplir los requerimientos referidos en la sección 3.2.

4.1. MODELO DE DATOS

Basados en el modelo de dominio descrito en la sección 3.1, se efecúa el modelo de entidad relación que contiene entidades básicas de la fig. 34.


RolUsuario
idRolUsuario Number(10)
NombreRol Variable characters(100)
DescripciónRol Variable characters(256)
Usuario_REF_RolUsuario
Usuario
NombreUsuario Variable characters(100)
ApellidoPaterno Variable characters(100)
ApellidoMaterno Variable characters(100)
Email Variable characters(256)
RUN Number(8)
Password Variable characters(256)
fechaIngresoUsuario Date
fechaActualizaciónUsuario Date
UsuarioEnProyecto_REF_Usuario
Cargo
idCargo Number(10)
NombreCargo Variable characters(100)
DescripciónCargo Variable characters(256)
UsuarioEnProyecto_REF_Cargo
UsuarioEnProyecto
idUsuarioEnProyecto Number(10)
fechaIngresoCargo Date
CambioCargoFlag Short integer
CargoEsVigente Short integer
UsuarioEnProyecto_REF_EstadoUsuarioEnProyecto
EstadoUsuarioEnProyecto
idEstadoUsuarioEnProyecto Number(10)
NombreEstadoUsuario Variable characters(100)
DescripciónEstadoUsuario Variable characters(256)


TipoProyecto
idTipoProyecto Number(10)
NombreTipoProyecto Variable characters(100)
DescripciónTipoProyecto Variable characters(256)
Proyecto_REF_TipoProyecto
Proyecto
idProyecto Number(10)
NombreProyecto Variable characters(100)
DescripcionProyecto Variable characters(256)
tieneFinanciamiento Short integer
esOpenSource Short integer
relacionadoConUSM Short integer
fechaEstadoProyecto Date
fechaCreacionProyecto Date
fechActualizaciónProyecto Date
códigoProyecto Variable characters(100)
Proyecto_REF_Estado
EstadoProyecto
idEstadoProyecto Number(10)
NombreEstadoProyecto Variable characters(100)
DescripciónEstadoProyecto Variable characters(256)
Reserva_REF_Proyecto
Reserva
idReserva Number(10)
DescripciónReserva Variable characters(256)
FechaInicio Date
FechaTérmino Date
BloqueInicio Integer
BloqueTérmino Integer
fechaCreacionReserva Date
fechaActualizacionReserva Date
Reserva_REF_EstadoReserva
EstadoReserva
idEstadoReserva Number(10)
NombreEstadoReserva Variable characters(100)
DescripciónEstadoReserva Variable characters(256)
Reserva_REF_Espacio


UnidadMedidaInsumo
idUnidadMedida Number(10)
NombreUnidad Variable characters(100)
DescripciónUnidad Variable characters(256)
Símbolo Variable characters(5)
Insumo_REF_UnidadMedidaInsumo
Insumo
idInsumo Number(10)
NombreInsumo Variable characters(100)
DescripciónInsumo Variable characters(256)
Cantidad Integer
fechaActualizacionInsumo Date
codigoInsumo Variable characters(100)
fechaIngresoInsumo Date
InsumoEnReserva_REF_Insumo
InsumoEnReserva
idInsumoEnReserva Number(10)
TipoEquipamiento
idTipoEquipamiento Number(10)
NombreTipoEquipamiento Variable characters(100)
DescripciónTipoEquipamiento Variable characters(256)
Equipamiento_REF_TipoEquipamiento
Equipamiento
idEquipamiento Numb
NombreEquipamiento Variab
DescripciónEquipamiento Variab
CódigoEquipamiento Variab
Observación Variab
fechaEstadoEquipamiento Date
fechaIngresoEquipamiento Date
fechaActualizacionEquipamiento Date
uipamiento_REF_Estado
EquipamientoEnReserva_REF_Equipamiento
EquipamientoEnReserva
idEquipamientoEnReserva Number(10)
entoEnReserva_REF_Reserva
Espacio
idEspacio Number(10)
NombreEspacio Variable characters(100)
DescripciónEspacio Variable characters(256)
fechaIngresoEspacio Date
fechaEstadoEspacio Date
codigoEspacio Variable characters(100)
fechaActualizacionEspacio Date

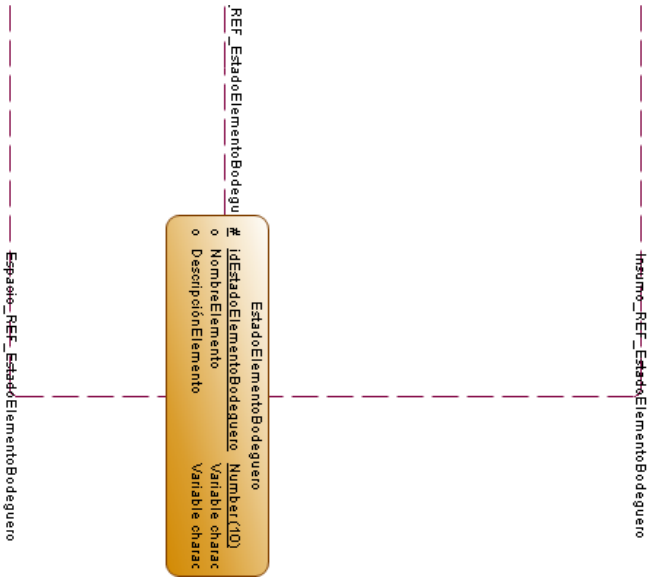


Figura 34: Modelado ER de los datos a manejar en Fab Lab.
Fuente: Elaboración Propia

Las entidades ”proyecto”, ”resultado”, ”reserva”, ”espacio” y ”equipamiento” se encuentran vinculadas a tablas relacionadas con losv “estados
bajo las cuales se pueden encontrar. Tener acceso por sistema a estos estados por cada entidad permite evitar la comprobación física o presencial de alguno de ellos.

La entidad “proyecto” tiene atributos tales como “tiene financiamiento?”, “es open source?” y “relacionado con USM?” ya que se pretente utilizar estos atributos como posibles criterios de priorización al momento de aprobar proyectos para la realización de reservas por parte del operador, facilitando la selección. El atributo “descripción” asume el rol de “resumen ejecutivo”, donde deben detallar objetivos del proyecto, entidad a cargo (puede ser una persona, departamento o empresa), y detalla específicamente cómo cumplen los criterios de priorización en caso de haber marcado alguna de las opciones en forma positiva.

La entidad “equipamiento” tiene atributo “observacion” que complementa “estado” con detalles, justificaciones o explicación complementaria.

Teniendo en consideración lo anterior, y los requisitos funcionales descritos en al sección 3.2, se efectúa el diagrama general de clases en la figura 35 que presenta las relaciones entre las diferentes entidades.

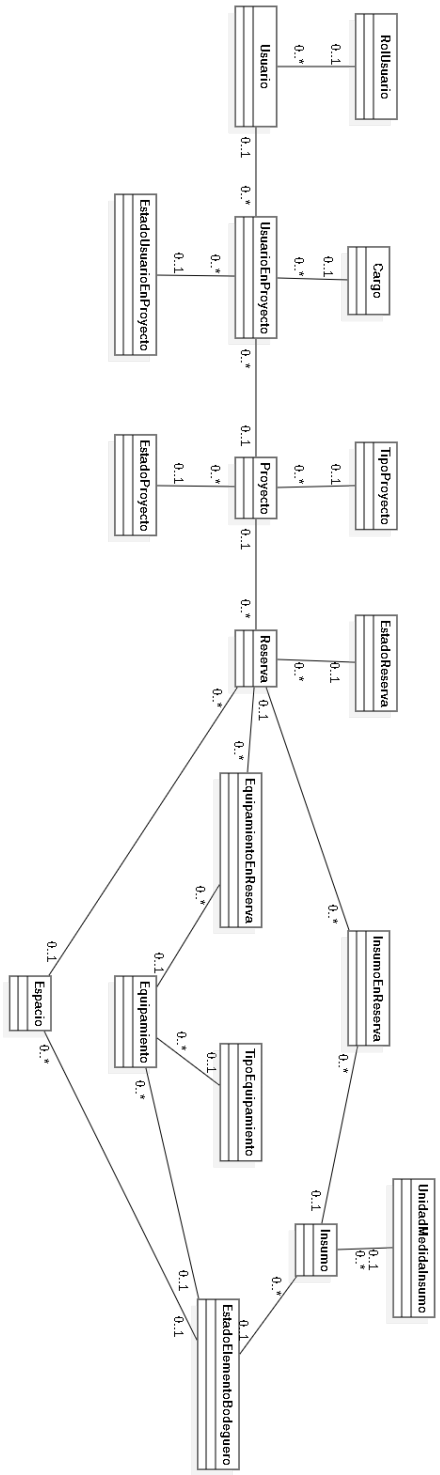
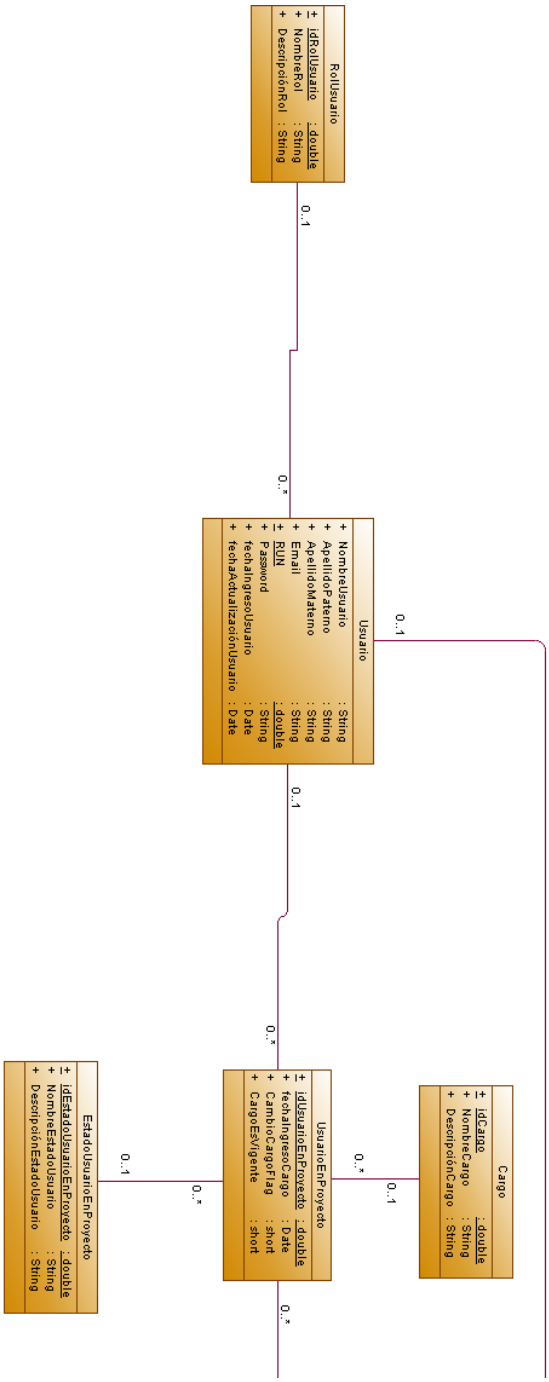
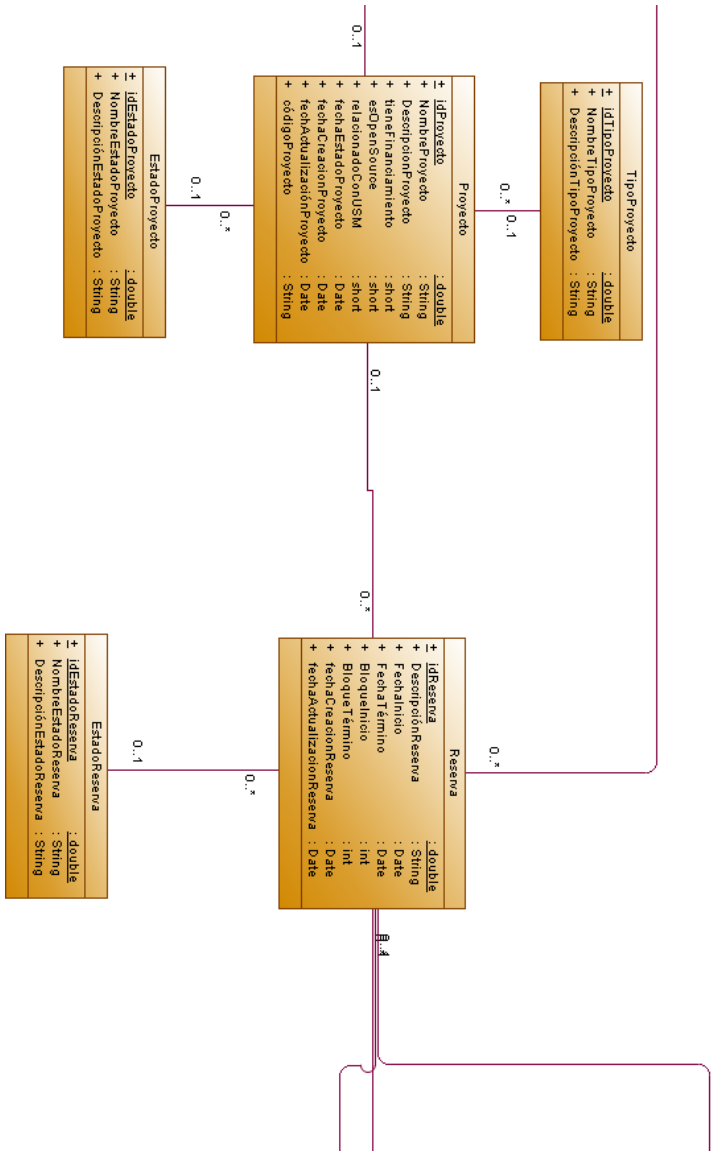


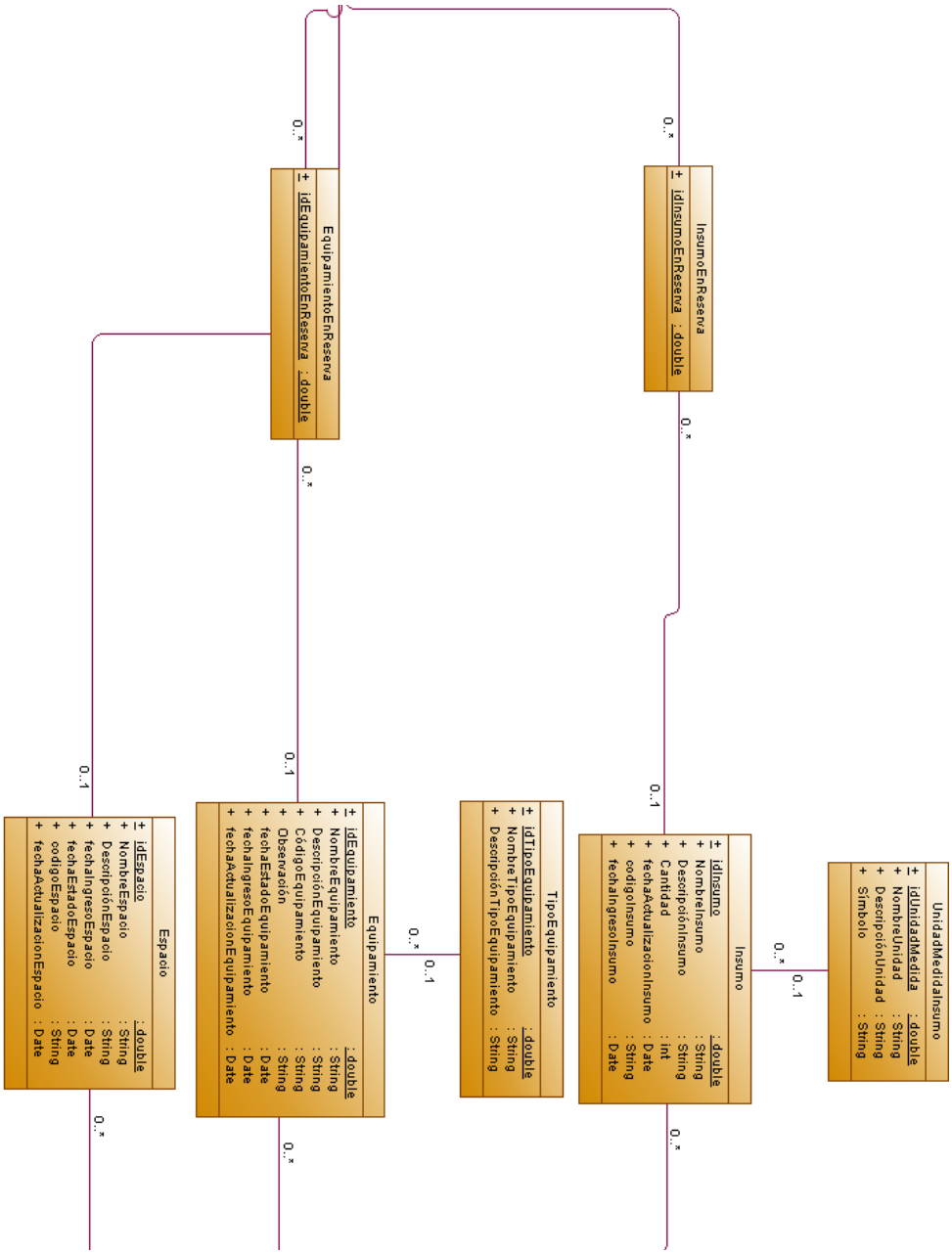
Figura 35: Modelo general de datos plataforma Fab Lab.
Fuente: Elaboración Propia

Dentro de los tipos de reservas, se consideran los préstamos, aquellas reservas realizadas por los participantes, evaluadas por los operarios, que involucran disponibilizar equipamiento en un espacio definido, el cual puede realizarse dentro de los espacios dispuestos por el Fab Lab. El equipamiento se compone de equipos, componentes electrónicos y herramientas, de las cuales al Fab Lab le interesa saber uso de los equipos, los cuales tienen un costo asociado a electricidad, insumos y tiempo de disponibilidad.

Más en detalle, el diagrama de clases con los atributos que presentan utilidad para las funcionalidades descritas se encuentran en la figura 36.


RolUsuario
+ idRolUsuario : double
+ NombreRol : String
+ DescripciónRol : String
0..1
0..*
Usuario
+ NombreUsuario : String
+ ApellidoPaterno : String
+ ApellidoMaterno : String
+ Email : String
+ RUN : double
+ Password : String
+ fechaIngresoUsuario : Date
+ fechaActualizaciónUsuario : Date
0..1
0..*
Cargo
+ idCargo : double
+ NombreCargo : String
+ DescripciónCargo : String
0..1
0..*
UsuarioEnProyecto
+ idUsuarioEnProyecto : double
+ fechaIngresoCargo : Date
+ CambioCargoFlag : short
+ CargoEsVigente : short
0..*
0..1
EstadoUsuarioEnProyecto
+ idEstadoUsuarioEnProyecto : double
+ NombreEstadoUsuario : String
+ DescripciónEstadoUsuario : String


TipoProyecto
+ idTipoProyecto : double
+ NombreTipoProyecto : String
+ DescripciónTipoProyecto : String
0..*
0..1
Proyecto
+ idProyecto : double
+ NombreProyecto : String
+ DescripciónProyecto : String
+ tieneFinanciamiento : short
+ esOpenSource : short
+ relacionadoConUSM : short
+ fechaEstadoProyecto : Date
+ fechaCreacionProyecto : Date
+ fechaActualizaciónProyecto : Date
+ códigoProyecto : String
0..1
0..*
EstadoProyecto
+ idEstadoProyecto : double
+ NombreEstadoProyecto : String
+ DescripciónEstadoProyecto : String
0..1
0..*
0..1
0..*
Reserva
+ idReserva : double
+ DescripciónReserva : String
+ FechaInicio : Date
+ FechaTérmino : Date
+ BloqueInicio : int
+ BloqueTérmino : int
+ fechaCreacionReserva : Date
+ fechaActualizacionReserva : Date
0..*
0..1
EstadoReserva
+ idEstadoReserva : double
+ NombreEstadoReserva : String
+ DescripciónEstadoReserva : String


InsumoEnReserva
+ idInsumoEnReserva : double
EquipamientoEnReserva
+ idEquipamientoEnReserva : double
UnidadMedidaInsumo
+ idUnidadMedida : double
+ NombreUnidad : String
+ DescripciónUnidad : String
+ Símbolo : String
Insumo
+ idInsumo : double
+ NombreInsumo : String
+ DescripciónInsumo : String
+ Cantidad : int
+ fechaActualizacionInsumo : Date
+ codigoInsumo : String
+ fechaIngresoInsumo : Date
TipoEquipamiento
+ idTipoEquipamiento : double
+ NombreTipoEquipamiento : String
+ DescripciónTipoEquipamiento : String
Equipamiento
+ idEquipamiento : double
+ NombreEquipamiento : String
+ DescripciónEquipamiento : String
+ CódigoEquipamiento : String
+ Observación : String
+ fechaEstadoEquipamiento : Date
+ fechaIngresoEquipamiento : Date
+ fechaActualizacionEquipamiento : Date
Espacio
+ idEspacio : double
+ NombreEspacio : String
+ DescripciónEspacio : String
+ fechaIngresoEspacio : Date
+ fechaEstadoEspacio : Date
+ codigoEspacio : String
+ fechaActualizacionEspacio : Date
0..*
0..1

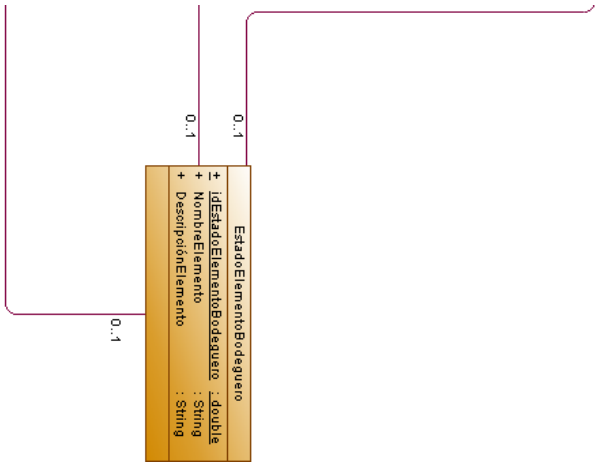
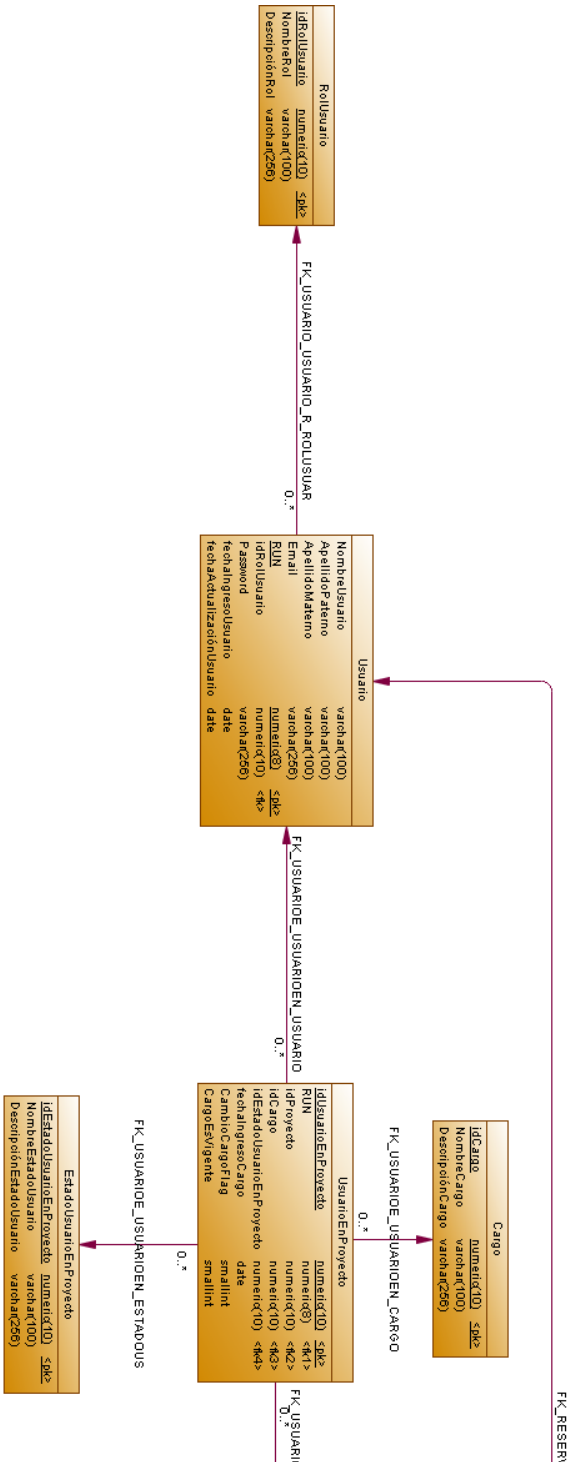
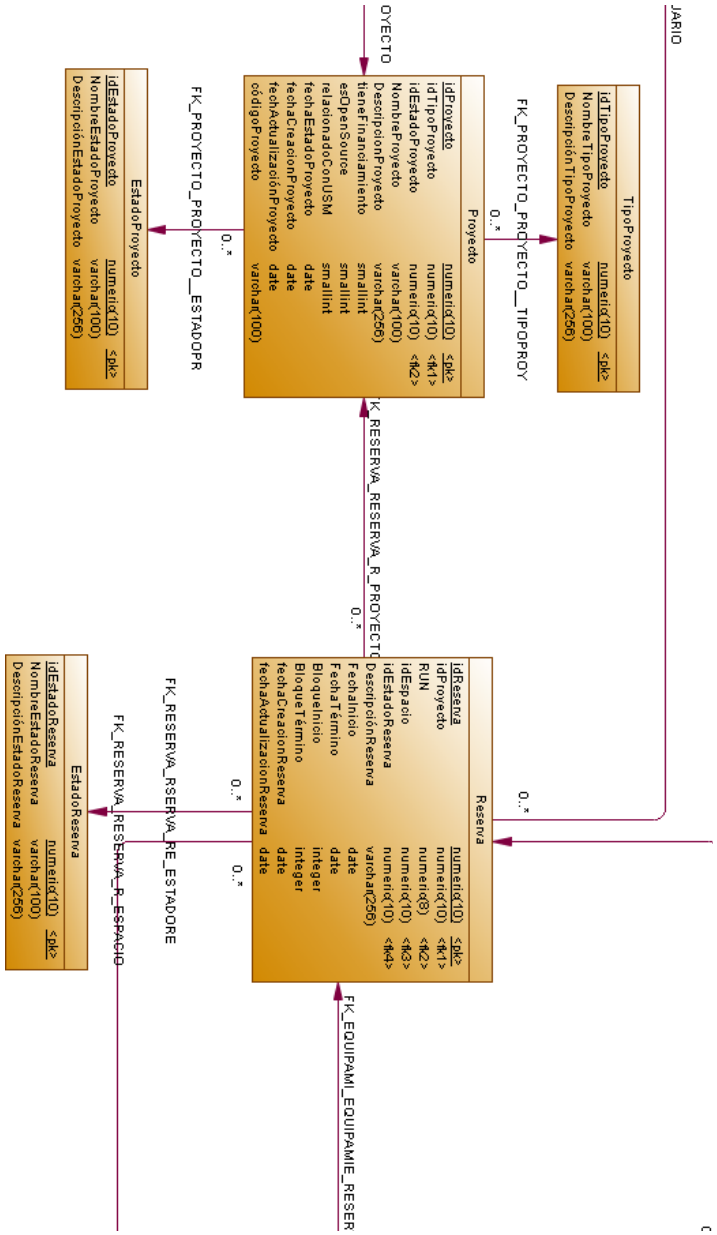


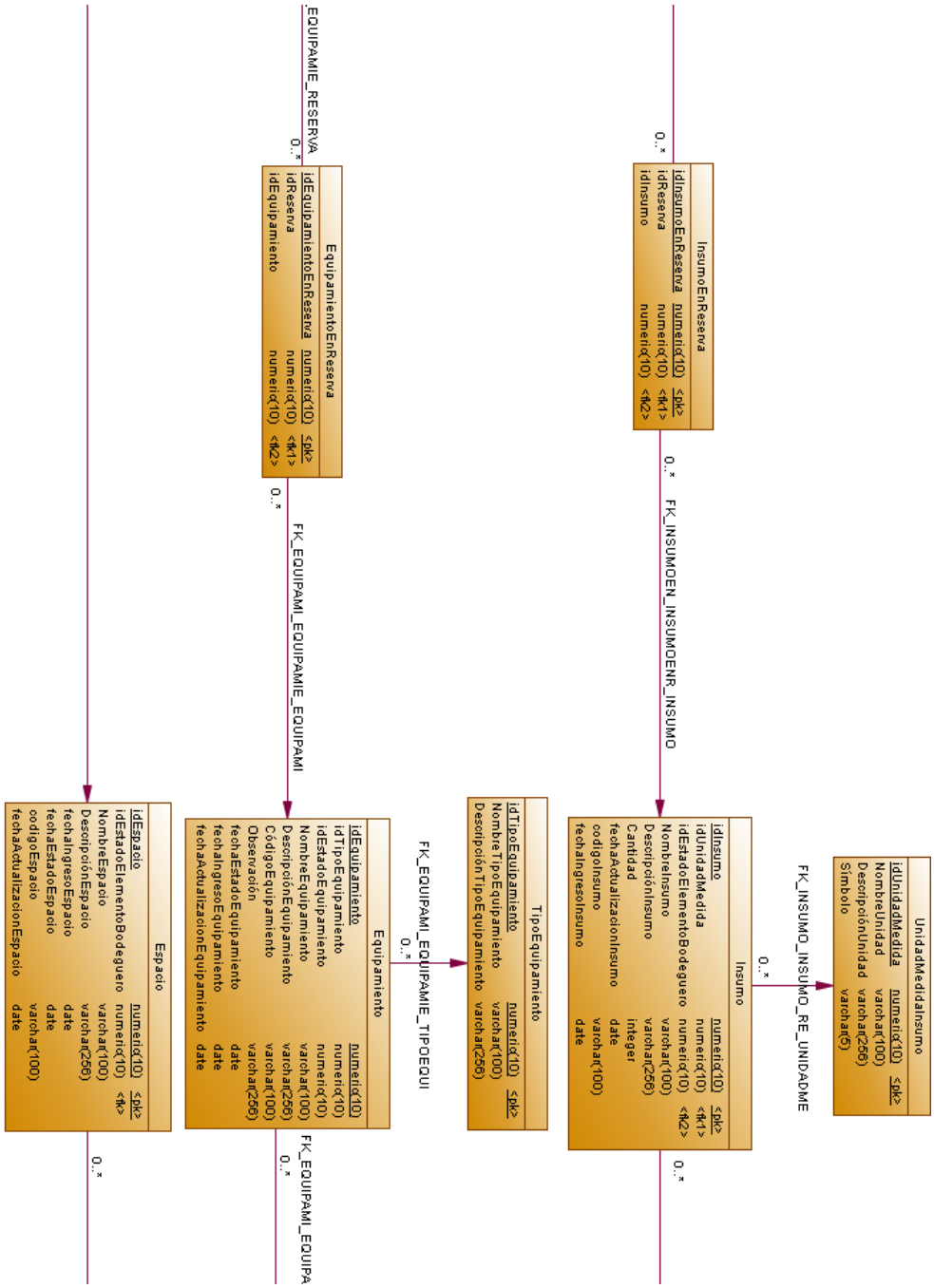
Figura 36: Diagrama de Clases de plataforma de Fab Lab.
Fuente: Elaboración Propia

4.2. MODELO FÍSICO

La figura 37 presenta cómo se organizan los datos para cumplir los requerimientos solicitados.


RolUsuario
idRolUsuario numeric(10) <pk>
NombreRol varchar(100)
DescripciónRol varchar(256)
FK_USUARIO_USUARIO_R_ROLUSUAR
0..*
Usuario
NombreUsuario varchar(100)
ApellidoPaterno varchar(100)
ApellidoMaterno varchar(100)
Email varchar(256)
RUN numeric(8) <pk>
idRolUsuario numeric(10) <fk>
Password varchar(256)
fechaIngresoUsuario date
fechaActualizaciónUsuario date
FK_USUARIOE_USUARIOEN_USUARIO
0..*
Cargo
idCargo numeric(10) <pk>
NombreCargo varchar(100)
DescripciónCargo varchar(256)
FK_USUARIOE_USUARIOEN_CARGO
0..*
UsuarioEnProyecto
idUsuarioEnProyecto numeric(10) <pk>
RUN numeric(8) <fk1>
idProyecto numeric(10) <fk2>
idCargo numeric(10) <fk3>
idEstadoUsuarioEnProyecto numeric(10) <fk4>
fechaIngresoCargo date
CambioCargoFlag smallint
CargoEsVigente smallint
FK_USUARIOE_USUARIOEN_ESTADOUS
0..*
EstadoUsuarioEnProyecto
idEstadoUsuarioEnProyecto numeric(10) <pk>
NombreEstadoUsuario varchar(100)
DescripciónEstadoUsuario varchar(256)
FK_RESER
FK_USUARI
0..*


JARIO
TipoProyecto
idTipoProyecto numeric(10) <pk>
NombreTipoProyecto varchar(100)
DescripciónTipoProyecto varchar(256)
FK_PROYECTO_PROYECTO__TIPOPROY
0..*
Proyecto
idProyecto numeric(10) <pk>
idTipoProyecto numeric(10) <fk1>
idEstadoProyecto numeric(10) <fk2>
NombreProyecto varchar(100)
DescripciónProyecto varchar(256)
tieneFinanciamiento smallint
esOpenSource smallint
relacionadoConUSM smallint
fechaEstadoProyecto date
fechaCreacionProyecto date
fechaActualizaciónProyecto date
códigoProyecto varchar(100)
OYECTO
FK_PROYECTO_PROYECTO__ESTADOPR
0..*
EstadoProyecto
idEstadoProyecto numeric(10) <pk>
NombreEstadoProyecto varchar(100)
DescripciónEstadoProyecto varchar(256)
K_RESERVA_RESERVA_R_PROYECTO
0..*
Reserva
idReserva numeric(10) <pk>
idProyecto numeric(10) <fk1>
RUN numeric(8) <fk2>
idEspacio numeric(10) <fk3>
idEstadoReserva numeric(10) <fk4>
DescripciónReserva varchar(256)
FechaInicio date
FechaTérmino date
BloqueInicio integer
BloqueTérmino integer
fechaCreacionReserva date
fechaActualizacionReserva date
0..*
FK_RESERVA_RSERVA_RE_ESTADORE
FK_RESERVA_RESERVA_R_ESPACIO
EstadoReserva
idEstadoReserva numeric(10) <pk>
NombreEstadoReserva varchar(100)
DescripciónEstadoReserva varchar(256)
FK_EQUIPAMI_EQUIPAMIE_RESER


UnidadMedidaInsumo
idUnidadMedida numeric(10) <pk>
NombreUnidad varchar(100)
DescripciónUnidad varchar(256)
Símbolo varchar(5)
FK_INSUMO_INSUMO_RE_UNIDADME
0..*
Insumo
idInsumo numeric(10) <pk>
idUnidadMedida numeric(10) <fk1>
idEstadoElementoBodeguero numeric(10) <fk2>
NombreInsumo varchar(100)
DescripciónInsumo varchar(256)
Cantidad integer
fechaActualizaciónInsumo date
codigoInsumo varchar(100)
fechaIngresoInsumo date
0..*
FK_INSUMOEN_INSUMOENR_INSUMO
0..*
InsumoEnReserva
idInsumoEnReserva numeric(10) <pk>
idReserva numeric(10) <fk1>
idInsumo numeric(10) <fk2>
0..*
TipoEquipamiento
idTipoEquipamiento numeric(10) <pk>
NombreTipoEquipamiento varchar(100)
DescripciónTipoEquipamiento varchar(256)
FK_EQUIPAMI_EQUIPAMIE_TIPOEQUI
0..*
Equipamiento
idEquipamiento numeric(10)
idTipoEquipamiento numeric(10)
idEstadoEquipamiento numeric(10)
NombreEquipamiento varchar(100)
DescripciónEquipamiento varchar(256)
CódigoEquipamiento varchar(100)
Observación varchar(256)
fechaEstadoEquipamiento date
fechaIngresoEquipamiento date
fechaActualizaciónEquipamiento date
FK_EQUIPAMI_EQUIPA
0..*
FK_EQUIPAMI_EQUIPAMIE_EQUIPAMI
0..*
EquipamientoEnReserva
idEquipamientoEnReserva numeric(10) <pk>
idReserva numeric(10) <fk1>
idEquipamiento numeric(10) <fk2>
EQUIPAMIE_RESERVA
0..*
Espacio
idEspacio numeric(10) <pk>
idEstadoElementoBodeguero numeric(10) <fk>
NombreEspacio varchar(100)
DescripciónEspacio varchar(256)
fechaIngresoEspacio date
fechaEstadoEspacio date
codigoEspacio varchar(100)
fechaActualizaciónEspacio date
0..*

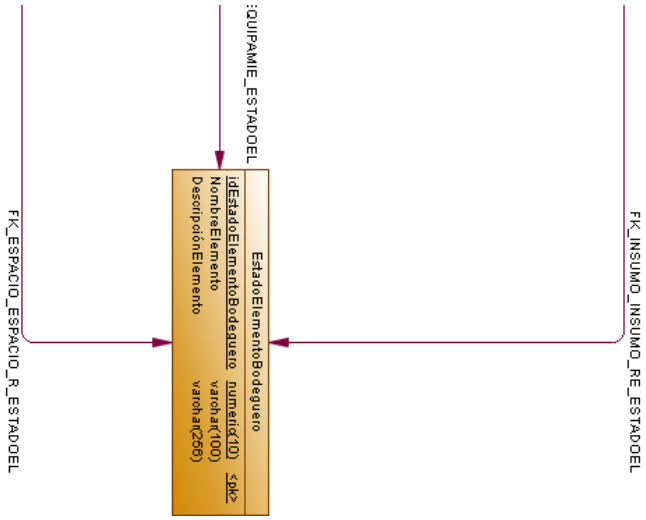


Figura 37: Modelo Físico de datos plataforma Fab Lab.
Fuente: Elaboración Propia

Para describir con exactitud las características que cada atributo debe tener, se detalla el siguiente diccionario de datos:

4.2.1. ROLUSUARIO

Contiene la lista de roles que puede tener un usuario y la descripción que lo resume. La tabla 21 contiene el detalle de los atributos correspondientes.

Campo	Tipo	Llave	Descripción
idRolUsuario	numeric(10)	Primaria	Identificador de rol.
NombreRol	varchar(100)	Atributo	Nombre de rol de usuario.
DescripcionRol	varchar(256)	Atributo	Descripción del rol.

Tabla 21: Tabla de atributos de los roles de usuario

4.2.2. USUARIO

Almacena los datos de toda persona que interactúe con las funcionalidades principales de la plataforma, independiente de su rol. La tabla 22 detalla las características importantes a almacenar.

Campo	Tipo	Llave	Descripción
RUN	numeric(8)	Primaria	Rol único nacional del usuario. Formato numérico. Valor único.
idRolUsuario	numeric(10)	Foránea	Identificador del rol del usuario en la plataforma.
NombreUsuario	varchar(100)	Atributo	Primer nombre de usuario.
ApellidoPaterno	varchar(100)	Atributo	Apellido paterno de usuario.
ApellidoMaterno	varchar(100)	Atributo	Apellido materno de usuario.
EMail	varchar(256)	Atributo	E-Mail de Usuario, necesario para ingresar. Valor único.
passwd	varchar(100)	Atributo	Contrasña del usuario, necesario para ingresar. Formato password.
fechaIngresoUsuario	date	Atributo	Fecha de ingreso del usuario.
fechaActualizacionUsuario	date	Atributo	Fecha de última actualización del usuario.

Tabla 22: Tabla de atributos de entidad Usuario

4.2.3. CARGO

Contiene la lista de cargos que puede tener un usuario dentro de un proyecto. La tabla 23 especifica cuales son los atributos de importancia para Fab Lab.

Campo	Tipo	Llave	Descripción
idCargo	numeric(10)	Primaria	Identificador de cargo.
NombreCargo	varchar(100)	Atributo	Nombre de cargo en proyecto.
DescripcionCargo	varchar(256)	Atributo	Descripción del cargo.

Tabla 23: Tabla de atributos de los cargos de un usuario en un proyecto

4.2.4. USUARIOENPROYECTO

Grupo relaciona los proyectos con los participantes, detallando si son responsables de éste o solo miembros y también el estado de aprobación de un participante cuando solicita integrarse a un proyecto. La tabla 24 expone cómo maneja la identificación de cargos dentro de los proyectos.

Campo	Tipo	Llave	Descripción
idUsuarioEnProyecto	numeric(10)	Primaria	Identificador único de usuario en un proyecto.
RUN	numeric(8)	Foránea	Identificador de usuario.
idProyecto	numeric(10)	Foránea	Identificador de proyecto.
idCargo	numeric(10)	Foránea	Identificador de Cargo.
idEstadoUsuarioEnProyecto	numeric(10)	Foránea	Identificador de estado del usuario en proyecto.
fecheIngresoCargo	date	Atributo	Fecha de ingreso del cargo al sistema.
cambioCargoFlag	smallint	Atributo	Indicador de registro en donde hubo cambio de cargo.
cargoEsVigente	smallint	Atributo	Indicador de registro con cargo vigente.

Tabla 24: Tabla de atributos de un usuario en un proyecto

4.2.5. ESTADOUSUARIOENPROYECTO

Contiene la lista de estados en los cuales puede estar un usuario dentro de un proyecto. La tabla 25 describe los campos necesarios para ello.

Campo	Tipo	Llave	Descripción
idEstadoUsuarioEnProyecto	numeric(10)	Primaria	Identificador de Estado de Usuario.
NombreEstadoUsuario	varchar(100)	Atributo	Nombre de estado del usuario en proyecto.
DescripcionEstadoUsuario	varchar(256)	Atributo	Descripción del estado de un usuario en un proyecto.

Tabla 25: Tabla de atributos de los estados de un usuario en un proyecto

4.2.6. TIPOPROYECTO

Contiene la lista de tipos de proyecto y sus descripciones. La tabla 26 muestra el contenido que debiese tener.

Campo	Tipo	Llave	Descripción
idTipoProyecto	numeric(10)	Primaria	Identificador de tipo de proyecto.
NombreTipoProyecto	varchar(100)	Atributo	Nombre del tipo de proyecto.
DescripcionTipoProyecto	varchar(256)	Atributo	Descripción del tipo de proyecto.

Tabla 26: Tabla de atributos de los tipos de proyecto

4.2.7. PROYECTO

Contiene las características principales de un proyecto, especificando ciertos aspectos de relevancia para el Fab Lab que le ayuda a filtrar y decidir aprobarlos. Los datos fundamentales que tiene se encuentran en la tabla 27.

Campo	Tipo	Llave	Descripción
idProyecto	numeric(10)	Primaria	Identificador de proyecto.
idTipoProyecto	numeric(10)	Foránea	Identificador del tipo de proyecto.
idEstadoProyecto	numeric(10)	Foránea	Identificador de estado del proyecto.
NombreProyecto	varchar(100)	Atributo	Nombre de proyecto.
DescripcionProyecto	varchar(256)	Atributo	Resumen ejecutivo de proyecto.
tieneFinanciamiento	smallint	Atributo	Establece si existe financiamiento o no.
esOpenSource	smallint	Atributo	Establece si tiene algún componente open source u open hardware.
relacionadoConUSM	smallint	Atributo	Establece si existe relación o beneficio para la comunidad USM.
fechaEstadoProyecto	date	Atributo	Fecha en que se establece estado del proyecto.
fechaCreacionProyecto	date	Atributo	Fecha en que se crea registro de proyecto.
fechaActualizacionProyecto	date	Atributo	Fecha en que se actualiza proyecto.
codigoProyecto	varchar(100)	Atributo	Código que identifica a un proyecto. Valor único.

Tabla 27: Tabla de atributos de entidad Proyecto

4.2.8. ESTADOPROYECTO

Contiene la lista de estados en los que puede estar un proyecto. Para ello, se enlistan los atributos en la tabla 28.

Campo	Tipo	Llave	Descripción
idEstadoProyecto	numeric(10)	Primaria	Identificador de estado del proyecto.
NombreEstadoProyecto	varchar(100)	Atributo	Nombre de estado del proyecto.
DescripcionEstadoProyecto	varchar(256)	Atributo	Descripción de estado del proyecto.

Tabla 28: Tabla de atributos de los estados de un proyecto

4.2.9. RESERVA

Tiene los datos esenciales relacionados con las reservas, los cuales se encuentran detallados en la tabla 29.

Campo	Tipo	Llave	Descripción
idReserva	numeric(10)	Primaria	Identificador de reserva.
idProyecto	numeric(10)	Foránea	Identificador de proyecto.
RUN	numeric(8)	Foránea	Identificador de usuario.
idEspacio	numeric(10)	Foránea	Identificador de Espacio.
idEstadoReserva	numeric(10)	Foránea	Identificador de estado de reserva.
DescripcionReserva	varchar(256)	Atributo	Características de la reserva.
fechaInicio	date	Atributo	Fecha de inicio de la reserva.
fechaTermino	date	Atributo	Fecha de término de la reserva.
bloqueInicio	integer	Atributo	Bloque de inicio de la reserva.
bloqueTermino	integer	Atributo	Bloque de término de la reserva.
fechaCreacionReserva	date	Atributo	Fecha de creación de la reserva.
fechaActualizacionReserva	date	Atributo	Fecha de actualización de la reserva.

Tabla 29: Tabla de atributos de entidad Reserva

4.2.10. ESTADORESERVA

Contiene la lista de estados en los que puede estar una reserva, de manera análoga a como ocurre en un proyecto. La tabla 30 efectúa el compendio de características a considerar.

Campo	Tipo	Llave	Descripción
idEstadoReserva	numeric(10)	Primaria	Identificador de estado de la reserva.
NombreEstadoReserva	varchar(100)	Atributo	Nombre de estado de la reserva.
DescripcionEstadoreserva	varchar(256)	Atributo	Descripción de estado de la reserva.

Tabla 30: Tabla de atributos de los estados de una reserva

4.2.11. INSUMOENRESERVA

Almacena los insumos presentes en una reserva. La tabla 31 resume las propiedades que deben registrarse.

Campo	Tipo	Llave	Descripción
idInsumoEnReserva	numeric(10)	Primaria	Identificador del insumo en la reserva.
idReserva	numeric(10)	Foránea	Identificador de la reserva.
idInsumo	numeric(10)	Atributo	Identificador del insumo.

Tabla 31: Tabla que registra la existencia de un insumo en una reserva

4.2.12. EQUIPAMIENTOENRESERVA

Almacena el equipamiento presente en una reserva. Para cumplir con este objetivo, los datos mencionados en la tabla 32 sirven de apoyo.

Campo	Tipo	Llave	Descripción
idEquipamientoEnReserva	numeric(10)	Primaria	Identificador del equipamiento en la reserva.
idReserva	numeric(10)	Foránea	Identificador de la reserva.
idEquipamiento	numeric(10)	Atributo	Identificador del equipamiento.

Tabla 32: Tabla que registra la existencia de un equipamiento en una reserva

4.2.13. UNIDADMEDIDAINSUMO

Almacena las unidades de medida de los diferentes insumos. En la tabla 33 se consolidan los aspectos necesarios para identificarlos.

Campo	Tipo	Llave	Descripción
idUnidadMedida	numeric(10)	Primaria	Identificador de la unidad de medida.
NombreUnidad	varchar(100)	Atributo	Nombre completo de la unidad de medida.
DescripcionUnidad	varchar(256)	Atributo	Descripción de la unidad de medida.
Simbolo	varchar(5)	Atributo	Símbolo de unidad de medida.

Tabla 33: Tabla que registra las unidades de medida de los insumos

4.2.14. INSUMO

Almacena las características generales de un insumo, cuyos valores a guardar se encuentran detallados en la tabla 34.

Campo	Tipo	Llave	Descripción
idInsumo	numeric(10)	Primaria	Identificador de la unidad de medida.
idUnidadMedida	numeric(10)	Foránea	Identificador de la unidad de medida.
idEstadoElementoBodeguero	numeric(10)	Foránea	Identificador del estado que se encuentra el insumo.
NombreInsumo	varchar(100)	Atributo	Nombre completo del insumo.
DescripcionInsumo	varchar(256)	Atributo	Descripción del insumo.
Cantidad	integer	Atributo	Cuantificación del insumo.
fechaIngresoInsumo	date	Atributo	Fecha en que ingresa por primera vez insumo.
fechaActualizacionInsumo	varchar(5)	Atributo	Fecha de actualización de registro del insumo.
codigoInsumo	varchar(100)	Atributo	Código interno de identificación de insumo.

Tabla 34: Tabla que registra propiedades de un insumo

4.2.15. TIPOEQUIPAMIENTO

Contiene la lista de tipos de equipamiento y sus descripciones. La tabla 35 y los atributos allí detallados son los requeridos por el Fab Lab.

Campo	Tipo	Llave	Descripción
idTipoEquipamiento	numeric(10)	Primaria	Identificador de tipo de equipamiento.
NombreTipoEquipamiento	varchar(100)	Atributo	Nombre del tipo de equipamiento.
DescripcionTipoEquipamiento	varchar(256)	Atributo	Descripción del tipo de equipamiento.

Tabla 35: Tabla de atributos de los tipos de equipamiento

4.2.16. ESTADOELEMENTOBODEGUERO

Describe los posibles estados en que puede estar un elemento a cargo del bodeguero, ya sea equipamiento, espacio o insumo. Aquellos aspectos de importancia se conocen en la tabla 36.

Campo	Tipo	Llave	Descripción
idEstadoElementoBodeguero	numeric(10)	Primaria	Identificador de tipo de equipamiento.
NombreEstadoElemento	varchar(100)	Atributo	Nombre de estado del elemento.
DescripcionEstadoElemento	varchar(256)	Atributo	Descripción del estado del elemento.

Tabla 36: Tabla de atributos de los estados de un elemento a cargo de bodeguero

4.2.17. EQUIPAMIENTO

Comprende aspectos generales de los equipos, herramientas y componentes electrónicos presentes en el Fab Lab. Las definiciones realizadas para describirlos se ubican en la tabla 37.

Campo	Tipo	Llave	Descripción
idEquipamiento	numeric(10)	Primaria	Identificador de equipamiento.
idTipoEquipamiento	numeric(10)	Foránea	Identificador del tipo de equipamiento.
idEstadoEquipamiento	numeric(10)	Foránea	Identificador del estado de equipamiento.
nombreEquipamiento	varchar(100)	Atributo	Nombre de equipamiento.
descripciónEquipamiento	varchar(256)	Atributo	Descripción de equipamiento.
codigoEquipamiento	varchar(100)	Atributo	Número de identificación de elemento de inventario para Fab Lab.
observacion	varchar(256)	Atributo	Descripción detallada del estado del elemento específico.
fechaEstadoEquipamiento	date	Atributo	Fecha de estado del equipamiento.
fechaIngresoEquipamiento	date	Atributo	Fecha de ingreso de equipamiento al sistema.
fechaActualizacionEquipamiento	date	Atributo	Fecha de actualización de equipamiento.

Tabla 37: Tabla de atributos de entidad Inventario

4.2.18. ESPACIO

Especifica espacios en donde se pueden efectuar las reservas. Estos espacios tienen sus atributos mencionados y descritos en la tabla 38.

Campo	Tipo	Llave	Descripción
idEspacio	numeric(10)	Primaria	Identificador de espacio.
idEstadoEquipamiento	numeric(10)	Foránea	Identificador del estado de equipamiento.
NombreEspacio	varchar(100)	Atributo	Nombre de espacio.
DescripcionEspacio	varchar(256)	Atributo	Características del espacio.
fechaIngresoEspacio	date	Atributo	Fecha de creación del espacio en el sistema.
fechaActualizacionEspacio	date	Atributo	Fecha de actualización del espacio en el sistema.
fechaEstadoEspacio	date	Atributo	Fecha en que se puso en cierto estado el espacio.
codigoEspacio	varchar(100)	Atributo	Identificador único de cierto espacio.

Tabla 38: Tabla de atributos de entidad Espacio

CAPÍTULO 5
CONCLUSIONES

El trabajo descrito en este documento cubre las necesidades esenciales que Fab Lab necesita, estableciendo las bases de una plataforma funcional al corto plazo. Se priorizan todas aquellas actividades que tengan directa relación con las operaciones llevadas a cabo dentro del Fab Lab, entendiéndose que representa el inicio del desarrollo de una plataforma que tendrá más características a medida que se requieran.
Los problemas que se buscan solucionar tienen relación con la falta de digitalización y estructuración de datos presente en sus actividades, poco uso de las tecnologías aplicables en las operaciones cotidianas y una mejora en el control de los recursos que actualmente maneja el Fab Lab. Las soluciones propuestas a estos problemas son el establecimiento de roles con labores específicas, una reestructuración de los principales procesos llevados a cabo en base a estos roles, y el desarrollo de una plataforma que apoye estas soluciones con un mejor manejo de los datos que se generan a partir de su uso.

El proceso de levantamiento de información, más allá del nivel técnico, requiere de una comunicación constante con el cliente, mucho trabajo presencial que permita conocer el rubro en el que se desenvuelven, los intereses principales en sus quehaceres, cuales son las necesidades informáticas que se presentan. Un aspecto no considerado que es relevante en el estudio de los problemas existentes y los aspectos a mejorar, son los procesos principales que se llevan a cabo de manera particular en la empresa. Cuando se analizan las tareas fundamentales, las personas encargadas de esas tareas, la periodicidad y las condiciones bajo las cuaes se realizan, se tiene un mayor entendimiento del rol que tiene la solución informática que se busca implementar. Estos factores determinan los lineamientos comunes para elaborar objetivos específicos con métricas cuantificables y acordadas en conjunto, las cuales son usadas para decretar el nivel de éxito del proyecto.

Las soluciones propuestas para este proyecto en particular tienen un carácter libre, con escasas restricciones sujetas a los valores del Fab Lab. sin embargo, el alcance escogido para este trabajo está limitado a las operaciones prácticas primordiales que se ejecutan, dando como valor una optimización del almacenamiento de datos actual, una mejora en el estudio del uso de sus recursos, y otorgando una potencial herramienta que abarque mayor funcionalidad en el tiempo. Para incentivar la cultura de comunidad y mentalidad "DIY", se recomienda un desarrollo mediante plantillas de trabajo, que permita utilizar lo que otros desarrolladores hayan realizado como base, y a la vez, personalizando según parámetros y requisitos internos del Fab Lab. Se efectúan estimaciones y metodologías de trabajo para la investigación, desarrollo e implementación de la solución, así como propuestas de herramientas de soporte para el desarrollo.

Para cumplir con los requisitos del sistema elaborados, se redefinen los procesos y las

maneras de efectuar las tareas cotidianas, teniendo en consideración la solución propuesta. Este análisis representa una asistencia importante de entendimiento tanto para el analista como para el cliente, ya que permite visualizar el impacto de la implementación de esta solución, proyecta de manera más clara los roles que debe tomar cada actor y las acciones que debe efectuar para utilizar la solución. Cabe señalar que este proceso es uno de los cuales más tiempo se invierte en este caso, principalmente debido al desconocimiento inicial de lo que se busca concretar en un etapa inicial y la falta de estandarización en sus tareas actuales.

El diseño de la solución a nivel de la estructura escogida para los datos representan un modelo inicial para cubrir las funcionalidades acordadas en el levantamiento, y está contemplada la ampliación acorde a las nuevas funcionalidades y datos nuevos que puedan obtenerse en un trabajo futuro.

5.1. TRABAJO FUTURO

El análisis y diseño de las funcionalidades vistas en este documento cubren a grandes rasgos los aspectos operacionales más importantes que se llevan a cabo en el Fab Lab, de manera que los principales actores involucrados se vean beneficiados de la plataforma a construír, o les entregue un valor primordial. Sin embargo, existen desafíos anexos a estos procedimientos que tienen influencia relevante sobre los objetivos específicos expuestos en la sección 1.3.

- **Finanzas**: Para optimizar el manejo, uso y adquisición de los recursos del Fab Lab, es necesario mantener un manejo de las adquisiciones y gastos que realiza.
- **Planificación de Proyectos**: Si el Fab Lab se proyecta como una iniciativa abierta a todo público, es necesario que exista instancias de capacitación, mejora de gestión, mentorías especializadas en equipamiento, entre otros, que ayuden a aquellos participantes que desarrollan un proyecto en emplear mejor los recursos que reserva del Fab Lab.
- **Gestión de Capacitaciones**: Para que la planificación de proyectos tenga una mayor tasa de éxito, la plataforma debe contar con funciones que mejoren las capacidades actuales de efectuar capacitaciones.
- **Migración de Datos**: Como característica adicional, se propone efectuar planificaciones enfocadas en ETL, de manera tal que facilite la actualización de tecnologías y dar soporte a la automatización y estandarización de los datos.

ANEXOS

Esta sección presenta material complementario que apoya la lectura de este documento.

5.1. NOTACIÓN BPMN

Para diagramar los procesos relacionados con el Fab Lab, se utiliza Modelo y Notación de Procesos de Negocio, el que ha resultado de fácil lectura y comprensión por parte de los stakeholders. La figura 38 presenta un compendio de la simbología utilizada.

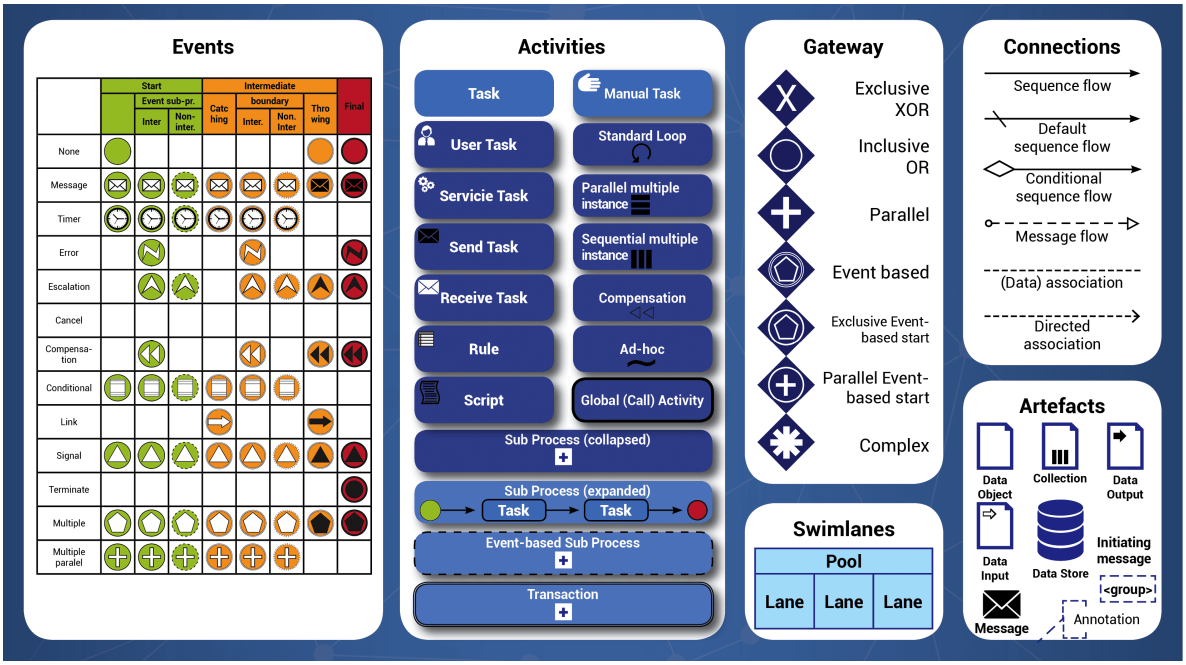


Figura 38: Notación BPMN 2.0
Fuente: http://www.diplomadosonline.com/elementos-de-la-notacion-bpmn-2-0/

5.2. MATRIZ DE TRAZABILIDAD

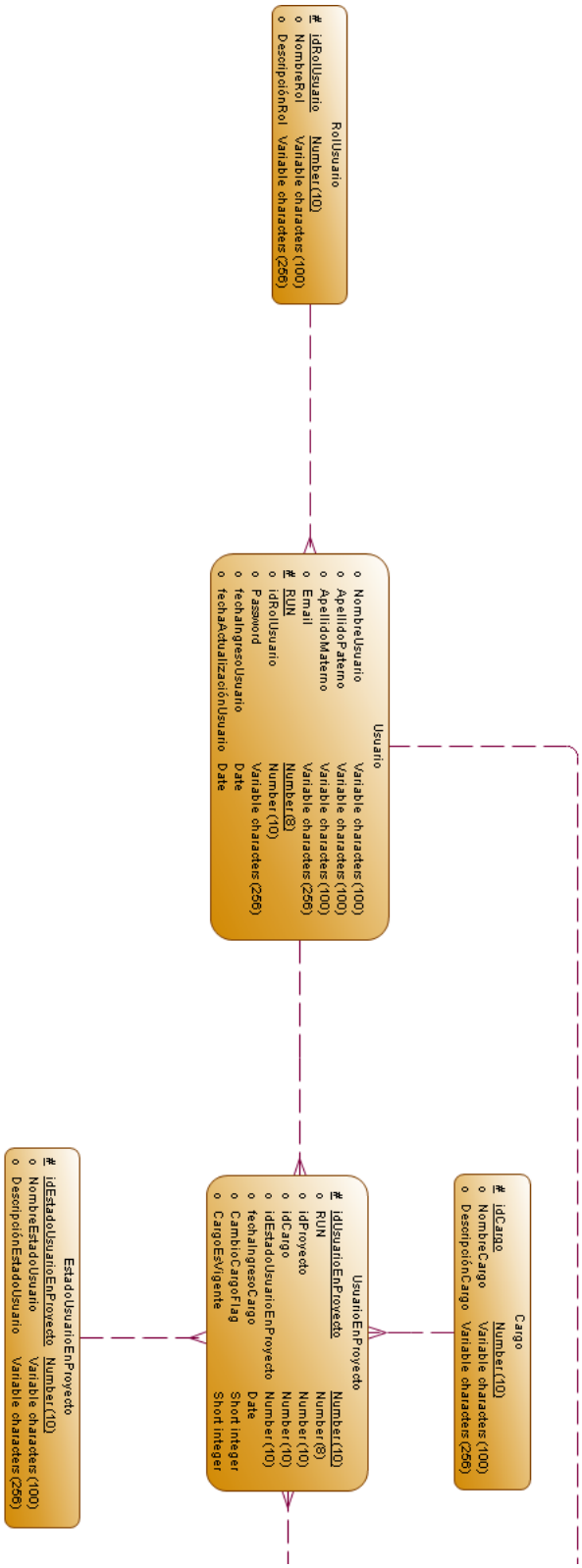
El seguimiento hecho a los diferentes requerimientos funcionales de la sección 3.2 en relación a los casos de uso ya detallados en la sección 3.4 se encuentra en la tabla 39.

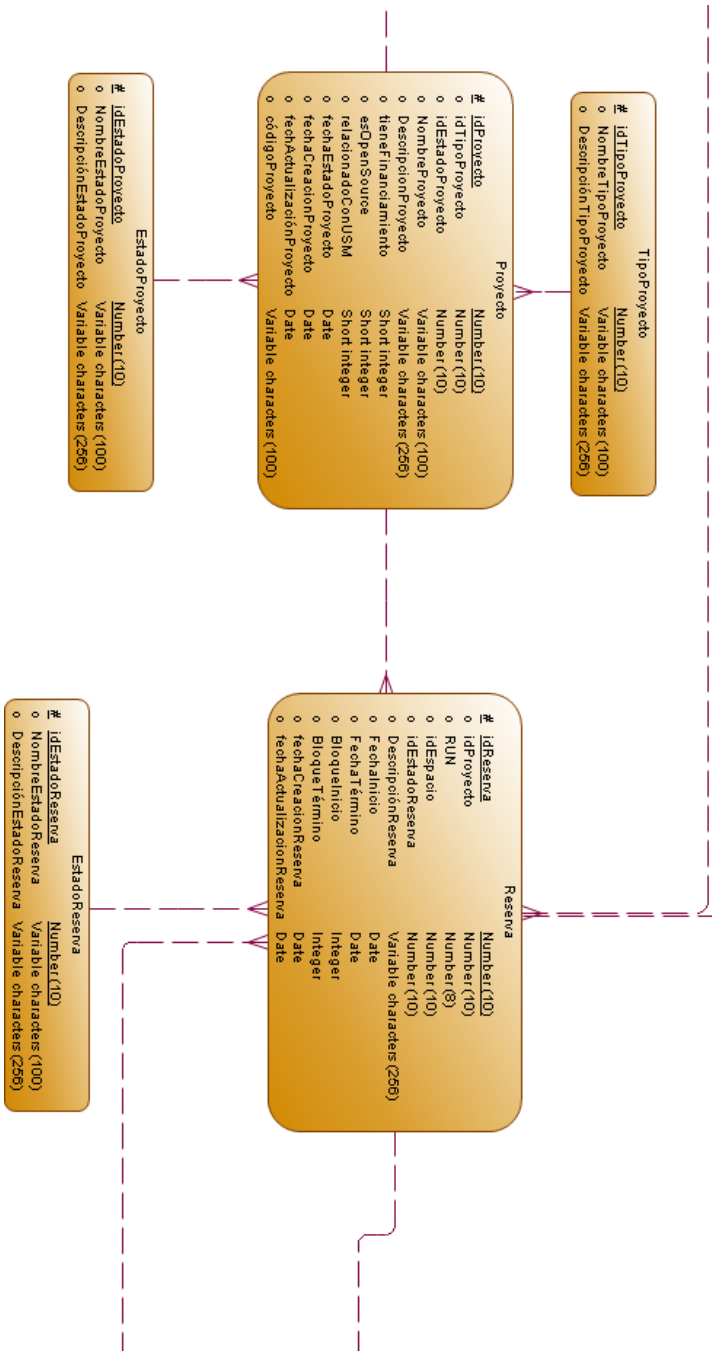
	RF-UNR01	RF-J01	RF-J02	RF-J03	RF-P01	RF-P02	RF-P03	RF-O01	RF-O02	RF-O03	RF-B01	RF-B02	RF-B03
CU-01	X												
CU-02		X											
CU-03								X					
CU-04		X											
CU-05					X								
CU-06			X										
CU-07						X							
CU-08									X				
CU-09											X		
CU-10											X		
CU-11												X	
CU-12							X						
CU-13				X									
CU-14													X
CU-15										X			

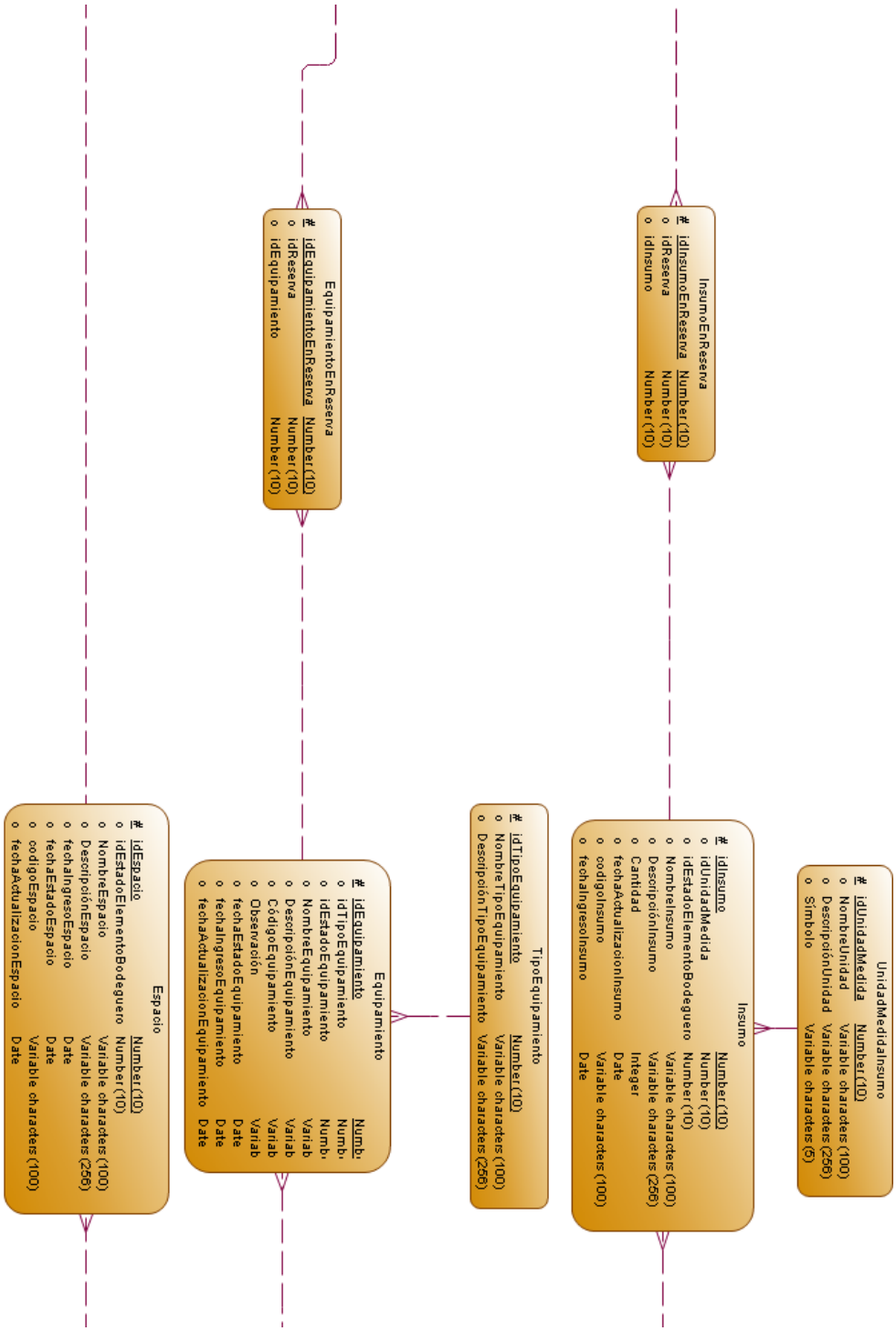
Figura 39: Matriz de trazabilidad de requerimientos
Fuente: Elaboraci'on Propia

5.3. MODELO LÓGICO

A continuación se detalla el modelo lógico generado a partir del modelo físico de la sección 4.2 de manera automática con la herramienta PowerDesigner.


RolUsuario
idRolUsuario Number(10)
NombreRol Variable characters(100)
DescripciónRol Variable characters(256)
Usuario
NombreUsuario Variable characters(100)
ApellidoPaterno Variable characters(100)
ApellidoMaterno Variable characters(100)
Email Variable characters(256)
RUN Number(8)
idRolUsuario Number(10)
Password Variable characters(256)
fechaIngresoUsuario Date
fechaActualizaciónUsuario Date
Cargo
idCargo Number(10)
NombreCargo Variable characters(100)
DescripciónCargo Variable characters(256)
UsuarioEnProyecto
idUsuarioEnProyecto Number(10)
RUN Number(8)
idProyecto Number(10)
idCargo Number(10)
idEstadoUsuarioEnProyecto Number(10)
fechaIngresoCargo Date
CambioCargoFlag Short integer
CargoEsVigente Short integer
EstadoUsuarioEnProyecto
idEstadoUsuarioEnProyecto Number(10)
NombreEstadoUsuario Variable characters(100)
DescripciónEstadoUsuario Variable characters(256)


TipoProyecto
idTipoProyecto Number(10)
NombreTipoProyecto Variable characters(100)
DescripciónTipoProyecto Variable characters(256)
Proyecto
idProyecto Number(10)
idTipoProyecto Number(10)
idEstadoProyecto Number(10)
NombreProyecto Variable characters(100)
DescripcionProyecto Variable characters(256)
tieneFinanciamiento Short integer
esOpenSource Short integer
relacionadoConUSM Short integer
fechaEstadoProyecto Date
fechaCreacionProyecto Date
fechaActualizaciónProyecto Date
códigoProyecto Variable characters(100)
EstadoProyecto
idEstadoProyecto Number(10)
NombreEstadoProyecto Variable characters(100)
DescripciónEstadoProyecto Variable characters(256)
Reserva
idReserva Number(10)
idProyecto Number(10)
RUN Number(8)
idEspacio Number(10)
idEstadoReserva Number(10)
DescripciónReserva Variable characters(256)
FechaInicio Date
FechaTérmino Date
BloqueInicio Integer
BloqueTérmino Integer
fechaCreacionReserva Date
fechaActualizacionReserva Date
EstadoReserva
idEstadoReserva Number(10)
NombreEstadoReserva Variable characters(100)
DescripciónEstadoReserva Variable characters(256)


UnidadMedidaInsumo
idUnidadMedida Number(10)
NombreUnidad Variable characters(100)
DescripciónUnidad Variable characters(256)
Símbolo Variable characters(5)
Insumo
idInsumo Number(10)
idUnidadMedida Number(10)
idEstadoElementoBodeguero Number(10)
NombreInsumo Variable characters(100)
DescripciónInsumo Variable characters(256)
Cantidad Integer
fechaActualizacionInsumo Date
codigoInsumo Variable characters(100)
fechaIngresoInsumo Date
InsumoEnReserva
idInsumoEnReserva Number(10)
idReserva Number(10)
idInsumo Number(10)
TipoEquipamiento
idTipoEquipamiento Number(10)
NombreTipoEquipamiento Variable characters(100)
DescripciónTipoEquipamiento Variable characters(256)
Equipamiento
idEquipamiento Numb
idTipoEquipamiento Numb
idEstadoEquipamiento Numb
NombreEquipamiento Variab
DescripciónEquipamiento Variab
CódigoEquipamiento Variab
Observación Variab
fechaEstadoEquipamiento Date
fechaIngresoEquipamiento Date
fechaActualizacionEquipamiento Date
EquipamientoEnReserva
idEquipamientoEnReserva Number(10)
idReserva Number(10)
idEquipamiento Number(10)
Espacio
idEspacio Number(10)
idEstadoElementoBodeguero Number(10)
NombreEspacio Variable characters(100)
DescripciónEspacio Variable characters(256)
fechaIngresoEspacio Date
fechaEstadoEspacio Date
codigoEspacio Variable characters(100)
fechaActualizacionEspacio Date

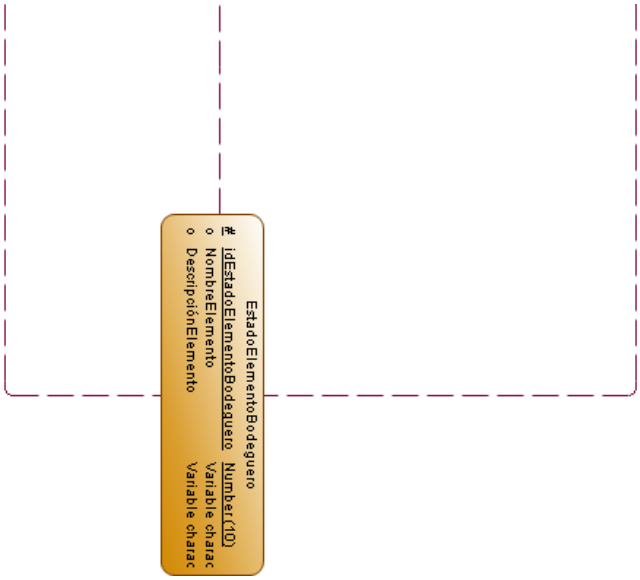


Figura 40: Modelo Lógico de datos plataforma Fab Lab.
Fuente: Elaboración Propia

REFERENCIAS BIBLIOGRÁFICAS

[FabLab, 2018] FabLab (2018). Historia. http://fablab.usm.cl/historia/.

[Gershenfeld, 2012] Gershenfeld, N. (2012). How to make almost anything, the digital fabrication revolution. http://cba.mit.edu/docs/papers/12.09.FA.pdf.

[Martí, 2012] Martí, J. L. (2012). Introducción a las bases de datos. Clase de Universidad.

[Rivera, 2018] Rivera, M. (2018). Fab lab utfsm | fablabs. https://www.fablabs.io/labs/fablabutfsm.

[UTFSM, 2017] UTFSM, D. (2017). Fablab - campus san joaquín. http://dea.usm.cl/2017/09/11/fablab-etapa-1-campus-san-joaquin/.

[UTFSM, 2018] UTFSM, F. (2018). Sobre nosotros. https://cl.linkedin.com/company/fablab-utfsm.

[Veltman, 2013] Veltman, N. (2013). Sql: The prequel (excel vs. databases). https://schoolofdata.org/2013/11/07/sql-databases-vs-excel/.