



UNIVERSIDAD TÉCNICA DE COTOPAXI
FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS
CARRERA DE INGENIERÍA ELECTROMECAÁNICA
PROPUESTA TECNOLÓGICA

“DESARROLLO DE UN MÓDULO VIRTUAL INMERSIVO PARA EL ENTRENAMIENTO DEL CONTROL DE PROCESOS DE FLUJO”

Propuesta Tecnológica presentada previo a la obtención
del Título de
INGENIEROS EN ELECTROMECAÁNICA

Autores:

Escobar Artieda Geovany Alexander

Imacaña Peñaloza Néstor Javier

Tutor:

Ing. Byron Paúl Corrales Bastidas M.Sc.

Latacunga - Ecuador

Marzo 2021



DECLARACIÓN DE AUTORÍA

Nosotros, **Escobar Artieda Geovany Alexander** e **Imacaña Peñaloza Néstor Javier** declaramos ser autores de la presente Propuesta Tecnológica:

“Desarrollo de un módulo virtual inmersivo para el entrenamiento del control de procesos de flujo”, siendo el Ing. Byron Paúl Corrales Bastidas M.Sc, tutor del presente trabajo; y eximimos expresamente a la Universidad Técnica de Cotopaxi y a sus presentes legales de posibles reclamos o acciones legales.

Además, certificamos que las ideas, conceptos, procedimientos y resultados vertidos en el presente trabajo investigativo, son de nuestra exclusiva responsabilidad.

Escobar Artieda Geovany Alexander

CC: 150080514-6

Imacaña Peñaloza Néstor Javier

CC: 180474898-4



AVAL DEL TUTOR DE PROYECTO DE TITULACIÓN

En calidad de Tutor del Trabajo de Investigación sobre el título:” **DESARROLLO DE UN MÓDULO VIRTUAL INMERSIVO PARA EL ENTRENAMIENTO DEL CONTROL DE PROCESOS DE FLUJO**” de Escobar Artieda Geovany Alexander e Imacaña Peñaloza Néstor Javier, de la Carrera de Ingeniería Electromecánica, considero que dicho Informe Investigativo cumple con los requerimientos metodológicos y aportes científicos-técnicos suficientes para ser sometidos a la evaluación del Tribunal de Validación de Proyecto que el Consejo Directivo de la Facultad de Ciencias de la Ingeniería y Aplicadas de la Universidad Técnica de Cotopaxi designe para su correspondiente estudio y calificación.

Latacunga, Marzo del 2021

Ing. Byron Paúl Corrales Bastidas M.Sc.
CC: 050234776-8



APROBACIÓN DEL TRIBUNAL DE TITULACIÓN

En calidad de Tribunal de lectores, aprueban el presente informe de Investigación de acuerdo a las disposiciones reglamentarias emitidas por la Universidad Técnica de Cotopaxi y por la Facultad de Ciencias de la Ingenierías y Aplicadas; por cuanto, los postulantes: Escobar Artieda Geovany Alexander con C.C 150080514-6 e Imacaña Peñaloza Néstor Javier con C.C 180474898-4 con el título de Proyecto de Titulación:

”DESARROLLO DE UN MÓDULO VIRTUAL INMERSIVO PARA EL ENTRENAMIENTO DEL CONTROL DE PROCESOS DE FLUJO”, han considerado las recomendaciones emitidas oportunamente y reúne los méritos suficientes para ser sometidos al acto de Sustentación de Proyecto.

Por lo antes expuesto, se autoriza realizar los empastados correspondientes, según la normativa institucional.

Latacunga, Marzo del 2021

Para constancia firman.

Lector 1

Ing. Cruz Panchi Luis Rolando
C.C: 050259517-6

Lector 2

Ing. Porras Reyes Jefferson Alberto
C.C: 070440044-9

Lector 3

Ing. Gallardo Molina Cristian Fabián
C.C: 050284769-2

AGRADECIMIENTO

Agradezco a Dios, a mi madre Anita y a mi abuelita Lucrecia, por brindarme su apoyo incondicional desde el inicio de mi carrera.

A mis hermanos Dayana y Rodney que nunca me dejaron solo y siempre estuvieron ahí cuando los necesitaba.

A la Universidad Técnica de Cotopaxi por permitir formarme como profesional y ser parte de la familia electromecánica. También quiero agradecer a mis compañeros, que siempre nos apoyamos mutuamente en los momentos más difíciles de la carrera, agradecer a nuestro tutor Ing. Byron Paúl Corrales Bastidas M.Sc por siempre estar atento y apoyándonos en nuestro proceso de titulación.

ALEXANDER

AGRADECIMIENTO

Quisiera agradecer a Dios, a mis padres y a mis hermanos que me apoyaron a lo largo de mi vida estudiantil, que nunca me dejaron solo, al apoyo incondicional que me brindan día a día para poder culminar mi carrera, también agradecer a mis compañeros que formaron parte de mi formación académica en la Universidad Técnica de Cotopaxi y a todos los docentes de Ingeniería Electromecánica que me impartieron sus conocimientos y su apoyo incondicional a nuestro tutor Ing. Byron Paúl Corrales Bastidas M.Sc. por haberme orientado con sus consejos y sabiduría a lo largo de mi formación académica.

NÉSTOR

DEDICATORIA

El presente trabajo de titulación está dedicado a toda mi familia en especial a mi madre Anita por ser mi inspiración y darme fuerza para obtener uno de los anhelos más deseados desde que entré a la Universidad Técnica de Cotopaxi. A todas las personas que me han apoyado a lo largo de toda mi carrera universitaria y compartieron su conocimiento conmigo.

ALEXANDER

DEDICATORIA

Este presente trabajo de investigación quiero dedicar a mis padres gracias a ellos y al apoyo incondicional de mis hermanos que me motivaron para terminar mi carrera profesional, gracias a ellos he podido cumplir todos las metas que me he propuesto, a mis compañeros que compartimos momentos agradables y que ellos han sido testigos del esfuerzo ya la dedicación que lo he realizado a lo largo de mi formación académica.

NÉSTOR

ÍNDICE DE CONTENIDO

DECLARACIÓN DE AUTORÍA	i
AVAL DEL TUTOR DE PROYECTO DE TITULACIÓN.....	ii
APROBACIÓN DEL TRIBUNAL DE TITULACIÓN	iii
AGRADECIMIENTO.....	iv
DEDICATORIA	vi
ÍNDICE DE CONTENIDO.....	viii
ÍNDICE DE FIGURAS.....	xi
ÍNDICE DE TABLAS	xiii
RESUMEN	xiv
ABSTRACT.....	xv
<i>AVAL DE TRADUCCIÓN</i>	xvi
1. INFORMACIÓN BASICA	1
2. DISEÑO INVESTIGATIVO DE LA PROPUESTA TECNOLÓGICA.....	2
2.1. TÍTULO DE LA PROPUESTA TECNOLÓGICA	2
2.2. TIPO DE PROPUESTA ALCANCE	2
2.3. ÁREA DE CONOCIMIENTO	2
2.4. SINOPSIS DE LA PROPUESTA TECNOLÓGICA.....	2
2.5. OBJETO DE ESTUDIO Y CAMPO DE ACCIÓN.....	2
2.5.1. Objeto de estudio.....	2
2.5.2. Campo de acción	2
2.6. SITUACIÓN PROBLÉMICA Y PROBLEMA.....	3
2.6.1. Situación problemática	3
2.6.2. Problema	3
2.6.3. Matriz causa efecto.....	4
2.7. HIPÓTESIS O FORMULACIÓN DE PREGUNTAS DIRECTRICES	4
2.8. OBJETIVOS	4
2.8.1. Objetivo general	4
2.8.2. Objetivo específico.....	4
2.9. DESCRIPCIÓN DE LAS ACTIVIDADES Y TAREAS PROPUESTAS CON OBJETIVOS ESTABLECIDOS	5
3. FUNDAMENTACIÓN CIENTÍFICO TÉCNICA	6
3.1. Análisis de trabajos precedentes.....	6
3.2. Control de procesos industriales.....	7
3.2.1. Tipos de procesos	8
3.2.1.1. Procesos auto regulados.....	8
3.2.1.2. Procesos integrantes	9
3.2.2. Procesos de flujo	10
3.2.2.1. Flujo volumétrico	10
3.2.2.2. Flujo másico	11

3.3.	Realidad Virtual	11
3.3.1.	Funcionamiento de los dispositivos de Realidad Virtual.	11
3.3.2.	Características básicas de la Realidad Virtual.....	12
3.3.3.	Tipos de Realidad Virtual.....	12
3.4.	Modelamiento 3D.....	12
3.4.1.	Softwares de Diseño CAD.....	13
3.4.2.	Diseño mecánico 3D.....	13
3.4.3.	Importación de modelamientos 3D a Unity	14
3.5.	Softwares de Realidad Virtual	15
3.5.1.	Concept 3D.....	15
3.5.2.1.	Unity 3D y Realidad Virtual	16
3.5.3.	Sistemas Inmersivos	16
3.5.3.1.	Casco de Realidad Virtual, Head-Mounted Display (HMD).....	16
3.5.3.2.	Caverna de inmersión, Cave Automatic Virtual Environment (CAVE).....	17
3.5.3.3.	Visión estereoscópica	18
3.6.	Hardware in the loop	19
3.6.1.	Desafíos y aplicaciones del sistema.....	19
3.7.	Sistema de Control de Procesos Industriales	20
3.7.1.	Control ON OFF.....	20
3.7.2.	Control PID	21
3.7.2.1.	Estructura del PID	21
3.7.3.	Control Moderno	23
3.7.4.	Control Fuzzy	23
3.7.5.	Control Predictivo	24
4.	METODOLOGÍA	25
4.1.	Implementación del control.....	27
4.1.1.	Diagrama P&ID de la estación de control de flujo	27
4.2.	Determinación de la función de transferencia.....	28
4.3.	Diseño simulación en LabVIEW	30
4.4.	Diseño de la planta en Inventor 2020	36
4.5.	Diseño del ambiente virtual en Unity 3D	38
4.6.	Integración del dispositivo Oculus Rift en Unity 3D.....	40
4.7.	Visualizador con Arduino.....	43
5.	ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS.....	44
5.1.	Respuesta del sistema ante la variación de Set point.	44
5.2.	Respuesta del sistema ante la presencia de perturbación.	46
5.3.	Análisis del entorno virtual.....	47
5.4.	Validación de resultado respecto al Modulo Real.....	49
6.	PRESUPUESTO Y ANÁLISIS DE IMPACTOS.....	52

6.1.	Presupuesto	52
6.1.1.	Materiales y equipos electrónicos.....	52
6.1.2.	Gastos de estructura y diseño de ingeniería del módulo virtual	53
6.1.3.	Gastos Varios	53
6.1.4.	Gastos directos totales	53
6.1.5.	Gastos indirectos totales.	54
6.1.6.	Gastos totales del módulo virtual.....	54
6.2.	Análisis de impacto	54
6.2.1.	Impacto tecnológico	54
6.2.2.	Impacto práctico	54
6.3.	Beneficiarios del proyecto.....	55
6.3.1.	Beneficiarios directos	55
6.3.2.	Beneficiarios indirectos	55
7.	CONCLUSIONES Y RECOMENDACIONES.....	55
7.1.	Conclusiones	55
7.2.	Recomendaciones.....	55
8.	REFERENCIAS.....	56
	ANEXOS	59

ÍNDICE DE FIGURAS

Figura 3.1	Control de un proceso asistido por un humano.	7
Figura 3.2	Control de un proceso automatizado.	8
Figura 3.3	Retardo lineal simple de caudal.	8
Figura 3.4	Capacidades de almacenamiento.	9
Figura 3.5	Proceso integrante.	10
Figura 3.6	Flujo másico de un fluido.	10
Figura 3.7	Como recrear imágenes estereoscópicas en 3D.	12
Figura 3.8	Modelado 3D.	14
Figura 3.9	Opciones de importación a Unity.	15
Figura 3.10	Gafas Oculus Rift de RV.	17
Figura 3.11	Sistema CAVE.	17
Figura 3.12	Ángulo de convergencia de dos imágenes monoculares.	18
Figura 3.13	Simulación de un sistema Hardware in the loop.	19
Figura 3.14	Relevador Electromecánico programable.	20
Figura 3.15	Diagrama de bloques.	21
Figura 3.16	Línea automatizada de ensamblajes de autos.	23
Figura 3.17	Lazo de control difuso.	23
Figura 3.18	Diagrama de control Fuzzy.	24
Figura 3.19	Estructura de un control predictivo.	25
Figura 4.1	Desarrollo del proyecto.	26
Figura 4.2	Sistema Didáctico.	27
Figura 4.3	Diagrama P&ID.	28
Figura 4.4	Control PID en LabVIEW.	30
Figura 4.5	Diagrama esquemático.	31
Figura 4.6	Setpoint.	31
Figura 4.7	Ganancias PID.	31
Figura 4.8	Función de transferencia.	32
Figura 4.9	Variación de válvula.	32
Figura 4.10	Respuesta del PID.	32
Figura 4.11	Nivel de tanque.	32
Figura 4.12	Estructura While.	33
Figura 4.13	Plant simulation.	33
Figura 4.14	Constantes.	34
Figura 4.15	Process Load.	34
Figura 4.16	Controlador PID.	35
Figura 4.17	Valores de ganancias PID.	35
Figura 4.18	Llenado y vaciado de tanques.	36

Figura 4.19 Soportes, tanques y bombas.	36
Figura 4.20 Ruteo de tuberías.....	37
Figura 4.21 Tubería completada.....	37
Figura 4.22 Norma seleccionada para tubería.	38
Figura 4.23 Diseño final de la planta.....	38
Figura 4.24 Entorno virtual en Unity 3D.....	39
Figura 4.25 Pantalla HMI.....	39
Figura 4.26 Visualización de objeto en primera persona.....	40
Figura 4.27 Gafas Oculus Rift.....	41
Figura 4.28 Aplicación de Oculus.....	41
Figura 4.29 Habilitar Oculus Rift.....	42
Figura 4.30 Paquete de Oculus en Unity 3D.	42
Figura 4.31 Tipo de orientación.	43
Figura 4.32 Orientación en el entorno virtual.....	43
Figura 4.33 Comunicación serial del visualizador de arduino.....	44
Figura 4.34. Indicadores led.....	44
Figura 5.1 Curva de funcionamiento del sistema en 10 L/min.	44
Figura 5.2 Curva de funcionamiento del sistema en 48 L/min.	45
Figura 5.3 Curva de funcionamiento del sistema en 70 L/min.	45
Figura 5.4 Perturbación en válvula estrangulada en 58°.....	46
Figura 5.5 Perturbación en válvula estrangulada en 90°.....	46
Figura 5.6 Perturbación en válvula estrangulada en 49°.....	47
Figura 5.7 Perturbación en válvula estrangulada en 0°.....	47
Figura 5.8 HMI del set point.....	48
Figura 5.9. Tanques de depósito y elevado.....	48
Figura 5.10. Movimiento de válvulas.....	48
Figura 5.11. Valores descendiendo a 0 lts/min.....	49
Figura 5.12. Tiempo de muestreo de 1 seg.....	50
Figura 5.13. Señales del módulo real y virtual.....	50
Figura 5.14. Señal del módulo real y virtual con set point 40.....	51
Figura 5.15. Señal del set point 60 y set point 80.....	51
Figura 5.16. Resultado de la comparación del módulo real y módulo inmersivo.....	52

ÍNDICE DE TABLAS

Tabla 2.1 Sistema de tareas por objetivos.....	5
Tabla 5.1 Resultados de porcentajes de sobre impulso y velocidad.....	45
Tabla 6.1 Materiales y equipos Electrónicos.	52
Tabla 6.2 Estructura y diseño de Ingeniería.	53
Tabla 6.3 Gastos varios.	53
Tabla 6.4 Gastos directos totales.	53
Tabla 6.5 Gastos indirectos totales.	54
Tabla 6.6 Gastos totales.	54

UNIVERSIDAD TÉCNICA DE COTOPAXI

FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS

TITULO: “DESARROLLO DE UN MÓDULO VIRTUAL INMERSIVO PARA EL ENTRENAMIENTO DEL CONTROL DE PROCESOS DE FLUJO”

Autores: Escobar Artieda Geovany Alexander

Imacaña Peñaloza Néstor Javier

RESUMEN

El análisis de Sistemas de Control es muy importante debido a que actualmente se desarrollan maquinas completamente automáticas por ende cada vez se hace más importante en la vida cotidiana y más aún en el sector industrial, en los procesos continuos de grandes empresas es difícil el análisis y la experimentación por tal motivo es necesario un entrenamiento mediante módulos didácticos.

En la actualidad en el Laboratorio de Instrumentación y Control no se dispone de un módulo virtual inmersivo de control de flujo en el cual se pueda experimentar con diferentes modelados matemáticos con el fin de que el estudiante analice y estudie el comportamiento de la variable flujo en un entorno virtual.

El presente proyecto pretende virtualizar un módulo didáctico de un proceso de flujo, que se desarrolla en dos etapas, la primera es modelar la planta en un software CAD posteriormente importar a Unity para crear un ambiente inmersivo que funcione con gafas de realidad virtual en donde la persona se involucra en el proceso a fin de analizar el comportamiento del mismo, la segunda etapa es el diseño de controladores para aplicar en el control virtual de flujo. Para ello se emplea el modelo matemático enlazando el software de instrumentación con el motor gráfico de Unity para el correcto funcionamiento, adicional se implementará un visualizador externo mediante Arduino Uno para observar datos del proceso de flujo.

El flujo se comporta de manera estable en todos los valores de referencia o setpoint que se añadió al sistema, las gafas de realidad virtual se comunican de manera efectiva con la computadora para la correcta inmersión. La comunicación Tcp/ip entre los dos softwares funciona correctamente y todo el sistema comparado con el modulo real tiene una similitud del 99.99 %.

Palabras claves: Realidad virtual, Inmersivo, Flujo, Control.

COTOPAXI TECHNICAL UNIVERSITY
FACULTY ENGINEERING AND APLIED SCIENCES

**TITLE: “DEVELOPMENT OF AN IMMERSIVE VIRTUAL MODULE FOR FLOW
PROCESS CONTROL TRAINING”**

Authors: Geovany Alexander Escobar Artieda

Néstor Javier Imacaña Peñaloza

ABSTRACT

The analysis of Control Systems is very important because fully automatic machines are currently being developed, therefore it is becoming more and more important in daily life and even more so in the industrial sector, in the continuous processes of large companies it is difficult to analyze and experimentation, for this reason requires training through didactic modules. Currently, the Instrumentation and Control Laboratory does not have an immersive virtual flow control module in which it is possible to experiment with different mathematical models in order for the student to analyze and study the behavior of the flow variable in a Virtual environment. This project aims to virtualize a didactic module of a flow process, which is developed in two stages, the first is to model the plant in CAD software, then import it into Unity to create an immersive environment that works with virtual reality glasses where the person is involved in the process in order to analyze its behavior, the second stage is the design of controllers to apply in the virtual flow control. For this, the mathematical model is used, linking the instrumentation software with the Unity graphic engine for correct operation, additionally an external display will be implemented using Arduino Uno to observe data from the flow process. The flux behaves in a stable way in all the reference values or set point that was added to the system, the virtual reality glasses communicate effectively with the computer for the correct immersion. The Tcp/Ip communication between the two software works correctly and the whole system comparing with the real module has a similarity of 99.99%.

Keywords: Virtual reality, Immersive, Flux, Control.



Universidad
Técnica de
Cotopaxi

CENTRO DE IDIOMAS

AVAL DE TRADUCCIÓN

En calidad de Docente del Idioma Inglés del Centro de Idiomas de la Universidad Técnica de Cotopaxi; en forma legal **CERTIFICO** que: La traducción del resumen del proyecto de investigación al Idioma Inglés presentado por los señores Egresados de la Carrera de Ingeniería Electromecánica de la Facultad de Ciencias de la Ingeniería y Aplicadas, **ESCOBAR ARTIEDA GEOVANY ALEXANDER** con cedula de identidad **1500805146** y **IMACAÑA PEÑALOZA NÉSTOR JAVIER** con cedula de identidad **1804748984**, cuyo título versa “**DESARROLLO DE UN MÓDULO VIRTUAL INMERSIVO PARA EL ENTRENAMIENTO DEL CONTROL DE PROCESOS DE FLUJO**” lo realizaron bajo mi supervisión y cumple con una correcta estructura gramatical del Idioma.

Es todo cuanto puedo certificar en honor a la verdad y autorizo a los peticionarios hacer uso del presente certificado de la manera ética que estimaren conveniente.

Latacunga, marzo del 2021

Atentamente.

Msc Lidia Rebeca Yugla Lema
DOCENTE CENTRO DE IDIOMAS
CC: 050265234-0

1803027935 Firmado digitalmente por
VICTOR HUGO ROMERO GARCIA
ROMERO GARCIA
GARCIA
Fecha: 2021.03.05 11:24:33 -05'00'

1. INFORMACIÓN BÁSICA

PROPUESTO POR:

Escobar Artieda Geovany Alexander

Imacaña Peñaloza Néstor Javier

TEMA APROBADO:

“Desarrollo de un módulo virtual inmersivo para el entrenamiento del control de procesos de flujo”

CARRERA:

Ingeniería Electromecánica.

DIRECTOR DEL PROYECTO DE TITULACIÓN:

Ing. Byron Paúl Corrales Bastidas M.Sc.

EQUIPO DE TRABAJO:

Sr. Geovany Alexander Escobar Artieda

Sr. Néstor Javier Imacaña Peñaloza

Ing. Byron Paúl Corrales Bastidas M.Sc

LUGAR DE EJECUCIÓN:

Zona 3, Provincia de Cotopaxi, Cantón Latacunga, Parroquia Eloy Alfaro, Barrio El Ejido, Universidad Técnica de Cotopaxi.

TIEMPO DE DURACIÓN DEL PROYECTO:

6 meses.

FECHA DE ENTREGA:

Marzo 2021.

EL PROYECTO RECAE SOBRE LA LÍNEA DE INVESTIGACIÓN:

De acuerdo a lo establecido por el departamento de investigación de la Universidad Técnica de Cotopaxi, línea 4 procesos industriales.

SUB LÍNEA DE INVESTIGACIÓN DE LA CARRERA:

Automatización, control y protección de sistemas electromecánicos.

TIPO DE PROPUESTA TECNOLÓGICA:

Es una propuesta tecnológica que busca desarrollar un módulo virtual inmersivo para el entrenamiento del control de procesos de flujo.

2. DISEÑO INVESTIGATIVO DE LA PROPUESTA TECNOLÓGICA

2.1. TÍTULO DE LA PROPUESTA TECNOLÓGICA

Desarrollo de un módulo virtual inmersivo para el entrenamiento del control de procesos de flujo.

2.2. TIPO DE PROPUESTA ALCANCE

Multipropósito: El módulo virtual está diseñado para la interacción virtual mediante software libre y realidad virtual, ayuda a los estudiantes con conocimientos prácticos en lo que se refiere a procesos de control de flujo.

2.3. ÁREA DE CONOCIMIENTO

La Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura (UNESCO) considera que el tema de propuesta tecnológica tiende a relacionarse con la ingeniería, industrial y construcción.

La UNESCO en la sub área de ingeniería y profesiones afines, el cual consta con: Dibujo técnico, mecánica, metalistería, electricidad, electrónica, telecomunicaciones, ingeniería energética y química, mantenimientos de vehículos, topografía.

2.4. SINOPSIS DE LA PROPUESTA TECNOLÓGICA

Se pretende realizar un módulo virtual inmersivo de control de flujo con la finalidad de introducir nuevas tecnologías de aprendizaje en los estudiantes utilizando softwares de programación gráfica para poder generar nuevas alternativas en el desarrollo de diferentes proyectos de instrumentación virtual. Mediante programas CAD se realiza el modelamiento 3D de un proceso de control de flujo que será similar a los existentes en el laboratorio de Automatización y Control de la Carrera de Ingeniería Electromecánica. El modelado realizado se importará directamente a un software de acceso libre en donde se realizará las programaciones gráficas y animaciones respectivas a fin de que cumpla con las mismas características de un proceso real. Una parte importante es el control del procesos para ello se implementa algoritmos de control clásicos para validar su funcionamiento. Mediante la inmersión se logra que el usuario pueda ingresar en la escena del proceso e interactuar de una manera adecuada con el mismo.

2.5. OBJETO DE ESTUDIO Y CAMPO DE ACCIÓN

2.5.1. Objeto de estudio

Módulo virtual inmersivo de control de un proceso flujo.

2.5.2. Campo de acción

Electrónica, Programación, Sistemas de control, Instrumentación, Microcontroladores.

2.6. SITUACIÓN PROBLÉMICA Y PROBLEMA

2.6.1. Situación problemática

El estudio de algunas variables que intervienen en un proceso industrial y que se relacionan con el control de procesos, se puede realizar mediante el empleo de módulos virtuales inmersivos que ayuden a entender el comportamiento natural de estos.

Los avances tecnológicos día a día son cada vez más sofisticados y de mayor demanda por parte de los seres humanos para facilitar tareas de la vida cotidiana, la realidad virtual es un tema sumamente interesante e importante pero bastante alejado de las personas, en Latinoamérica la realidad virtual es ajena a los avances tecnológicos y por ende cuesta acostumbra a entender como esta herramienta aportará a la educación[1].

La realidad virtual ofrece muchos recursos e implica menos costos y riesgos tanto para los equipos como para las personas los cuales no se necesitan equipos físicos a menudo costosos, ya que pueden repetir el entrenamiento tantas veces como sea necesario[2].

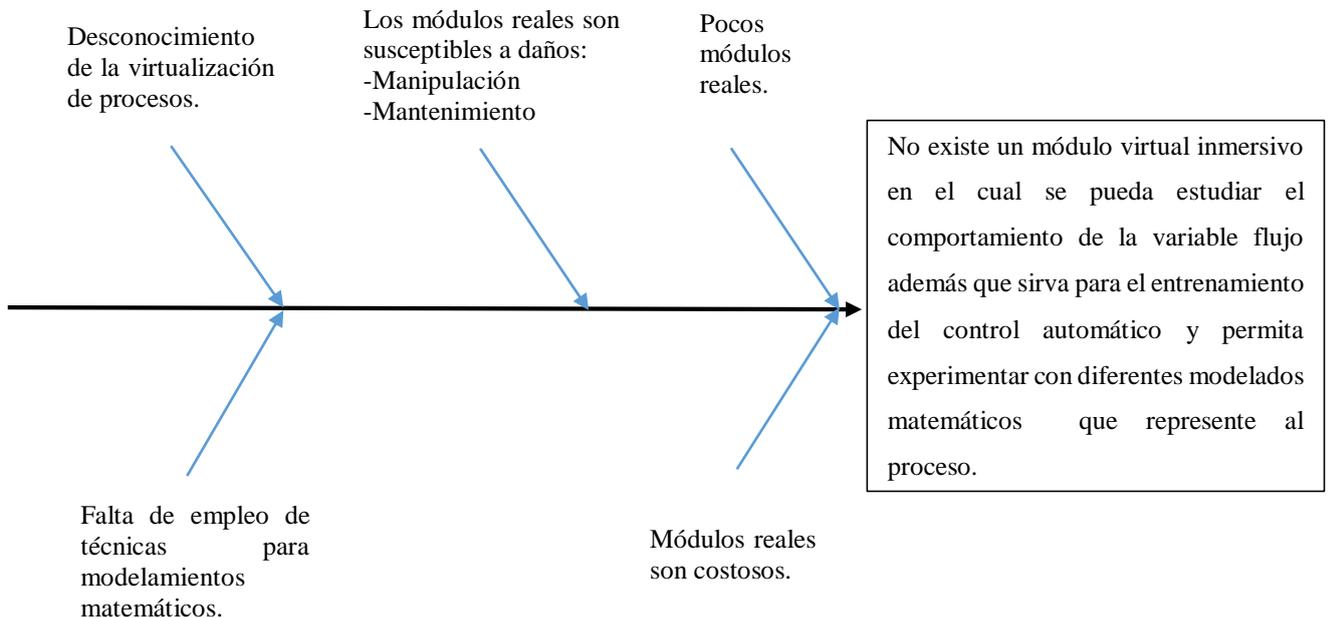
Los procesos más utilizados por las industrias son las de fluidos debido a que se encuentran frecuentemente por ende siempre deben estar medidos y controlados en todo momento. El agua es el fluido más utilizado por las industrias debido a su fácil obtención y es muy utilizado en todos los procesos industriales.

Actualmente no se dispone de módulos virtuales inmersivos de entrenamiento para el análisis y control de la variable flujo en el laboratorio de Ingeniería Electromecánica los cuales podrían ayudar a los estudiantes a realizar prácticas de laboratorio cuando los módulos físicos no estén disponibles, de igual forma cuando la máquina entra en mantenimiento o sufre daños de sus materiales por el constante uso se reducen aún más las posibilidades.

2.6.2. Problema

No existe un módulo virtual inmersivo en el cual se pueda estudiar y analizar el comportamiento de la variable flujo además que sirva para el entrenamiento del control automático y permita experimentar con diferentes modelados matemáticos que represente al proceso.

2.6.3. Matriz causa efecto



2.7. HIPÓTESIS O FORMULACIÓN DE PREGUNTAS DIRECTRICES

El desarrollo de un módulo virtual inmersivo permitirá el análisis del control automático del proceso de flujo, así como también la experimentación para determinar el modelo matemático de la planta.

2.8.OBJETIVOS

2.8.1. Objetivo general

Desarrollar un módulo virtual inmersivo para el entrenamiento del control de procesos de flujo utilizando softwares de animación gráfica.

2.8.2. Objetivo específico

- Investigar las características relevantes que intervienen en el funcionamiento de un proceso de flujo.
- Obtener la función de transferencia de un proceso de flujo.
- Desarrollar el entorno virtual de un proceso de flujo que cumpla con las características operativas de un proceso real.
- Demostrar el funcionamiento del módulo virtual inmersivo.

2.9. DESCRIPCIÓN DE LAS ACTIVIDADES Y TAREAS PROPUESTAS CON OBJETIVOS ESTABLECIDOS

Tabla 2.1 Sistema de tareas por objetivos.

OBJETIVOS	ACTIVIDAD	RESULTADOS	MEDIO DE VERIFICACIÓN
Investigar las características relevantes que intervienen en el funcionamiento de un proceso de flujo.	Recolección de información en fuentes bibliográficas y virtuales.	Análisis de proceso de flujo.	Artículos científicos, Libros y Páginas web.
	Revisión del funcionamiento de un proceso de flujo	Datos obtenidos, Información sobre el proceso que tiene un módulo de control de flujo.	Artículos científicos, Libros y Páginas web.
Obtener la función de transferencia de un proceso de flujo.	Realizar la conexión de todos los instrumentos para poder dar señales de entrada al módulo mediante el variador.	Conexión de instrumentos.	Base de datos.
	Recopilar todos los datos obtenidos para luego enviar a Matlab y con el comando systemidentification encontrar la función de transferencia.	Datos de flujo litros por minuto.	Base de datos.
Desarrollar el entorno virtual de un proceso de flujo que cumpla con las características operativas de un proceso real.	Modelamiento 3D del proceso de flujo a virtualizarse.	Modelado 3D.	Software CAD.
	Diseño de un sistema de control de proceso de flujo virtual.	Desarrollo de algoritmos de control.	Programación software de control de sistemas.
Demostrar el funcionamiento del módulo virtual inmersivo.	Pruebas experimentales de funcionamiento del prototipo.	Módulo externo de visualización de datos.	Módulo funcionando.
	Análisis del correcto funcionamiento del módulo inmersivo.	Correcta comunicación de las gafas de realidad virtual.	Módulo virtual de control de flujo.
	Evaluar la correcta programación para que el funcionamiento sea similar al módulo real.	Correcto funcionamiento.	Módulo virtual de control de flujo, Informe final.

3. FUNDAMENTACIÓN CIENTÍFICO TÉCNICA

3.1. Análisis de trabajos precedentes

Vergara J, Molina R, Machado C, [3] mencionan que el control PID permite optimizar el control sobre la velocidad de la bomba para lograr estabilizar el sistema con un error cercano a cero y por ello se aconseja utilizarlo en todos los sistemas de precisión, en el proceso de control de caudal es necesario optar por el lazo de control PID debido al requerimiento de exactitud para laboratorios.

Según Diego Aguiar Armada [4] menciona que Unity es uno de los softwares más completos para la creación de entornos 3D que cuenta con su propio sistema de animación y motores de física para las gráficas 2D y 3D que es muy usado en la industria de los videojuegos, en la actualidad este tipo de programas son usados en otras áreas como la Arquitectura y la Ingeniería por su capacidad de crear sistemas inmersivos virtuales.

Simulación inmersiva con realidad aumentada [5] concluye que para una mejor sensación de inmersión se necesita elementos o dispositivos con características de imagen estereoscópica como los HMD (Head Mounted Display) que en realidad son los cascos y gafas de RV, estos contienen sensores que captan los diferentes movimientos del cuerpo produciendo entradas en el ambiente virtual, así mismo se reciben respuestas que son generadas por el sentido de la persona.

Desarrollo y caracterización de un sistema de control P&ID virtual para los procesos de control de nivel y flujo de un tanque de agua [6] concluye que entre los diferentes sistemas de control que existen para procesos industriales el más completo hasta la actualidad es el P&ID porque contiene constantes como el proporcional que crea mediante una ganancia, una acción proporcional al error, el integral que genera señales crecientes y decrecientes obteniendo error cero en estado estacionario y el derivativo que mejora la estabilidad de control de lazo cerrado extrapolando el error. Todo este conjunto de constantes le hacen al control P&ID el más confiable de todos por su elevada precisión.

Diseño y construcción de un sistema didáctico de control de caudal [7] menciona que el flujo en las industrias es una de las variables más controladas por ende se han venido desarrollando varios sistemas para poder medirlo y controlarlo, actualmente existen varios sensores pero cada uno tiene características diferentes y deben ser escogidos de manera correcta teniendo en cuenta varios factores como tipo de fluido, el rango a medir, precisión presión precio etc.

Según Andrade Cedeño Rogger [8] menciona que la medición de fluidos como gases y

líquidos son frecuentes en los procesos de las industrias, existen dos tipos de medidores los volumétricos que son los más utilizados por realizar mediciones generales y los másicos que se los utiliza para aplicaciones que requieran mayor exactitud.

3.2. Control de procesos industriales

Los procesos de control industrial usualmente requieren ser controlados simultáneamente, dentro de estos se tiene: químicos, farmacéuticos biotecnológicos, petroquímicos, etc. El propósito es poder controlar las diferentes variables, como la presión el caudal y el flujo mediante lazos de control, el cual permite comparar corregir y cumplir con la finalidad del proceso designado, en la actualidad son de gran ayuda para las industrias, el trabajo que era realizado o supervisado por un humano ahora son remplazados por computadoras.

En la figura 3.1 se puede observar un sistema de control de flujo el cual está siendo controlada por una persona la que se encarga de realizar varias actividades como: sus ojos que están cumpliendo la función de un sensor, las manos se las tiene como actuador para poder girar la válvula de control según sea lo indicado, lo puede realizar tanto para abrir y cerrar la válvula para que el fluido circule o se detenga[9].

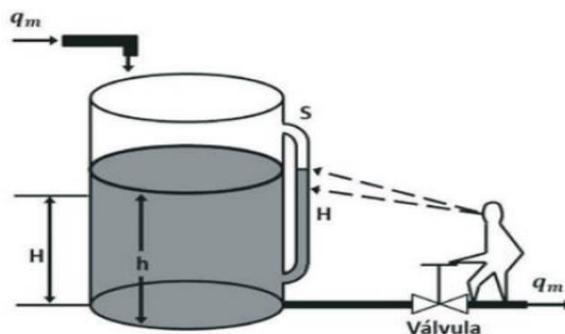


Figura 3.1 Control de un proceso asistido por un humano.

Fuente: [9].

En la figura 3.2 se puede observar un sistema de control automatizado de flujo. El proceso es controlado mediante sensores el cual reemplaza al hombre, es utilizado para el control del llenado del tanque el cual envía una señal al actuador que este procederá a abrir o cerrar la válvula.

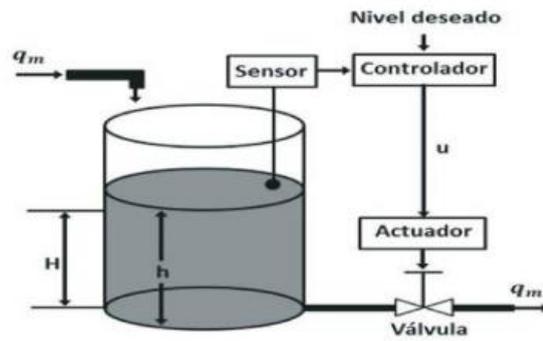


Figura 3.2 Control de un proceso automatizado.

Fuente: [9].

3.2.1. Tipos de procesos

Los tipos de procesos se caracterizan principalmente por su dinámica, uno es denominado control de lazo abierto y el otro en lazo cerrado, los procesos industriales existentes son; auto regulados o integrantes[10].

3.2.1.1. Procesos auto regulados

El proceso autorregulado muestran un cambio brusco en la señal de entrada como se muestra en la figura 3.3 que en operación normal sería la entrada que recibirá el controlador proveniente del proceso, se aprecia que la salida a alcanzado un nuevo punto el cual se mantiene constante[11].

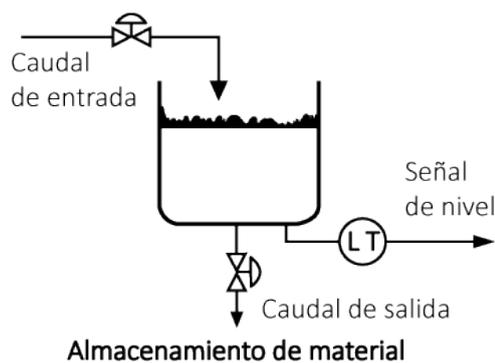


Figura 3.3 Retardo lineal simple de caudal.

Fuente: [12].

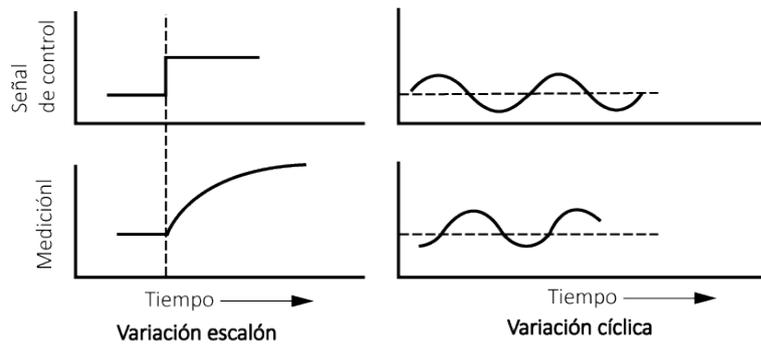


Figura 3.4 Capacidades de almacenamiento.

Fuente: [12].

$$G_m = (S) = \frac{K_m * e^{-s\tau_m}}{1+sT_m} \quad (3.1)$$

Dónde:

La ganancia de K_m se obtiene a partir de;

$$K_m = \frac{O_2 - O_1}{I_2 - I_1} \quad (3.2)$$

La constante de tiempo T_m se obtiene a partir de;

$$T_m = T_2 - T_1 \quad (3.3)$$

El tiempo muerto τ_m se obtiene a partir de;

$$\tau_m = T_3 - T_2 \quad (3.4)$$

3.2.1.2. Procesos integrantes

Los procesos integrativos del controlador de lazo abierto la variable de control es un conductor de modo deslizante, se presenta cuando tiene un desbalance de energía de flujo de entrada y la variable de proceso son idealmente controlados por una acción netamente proporcional tiende a subir o a bajar y se moverá de una forma lineal al cual se lo conoce como un proceso integrante. Depende de la cantidad de acción proporcional del tiempo de retardo del sistema y del ruido[13].

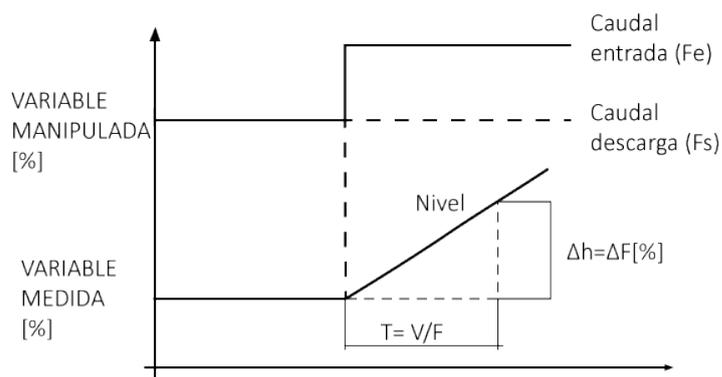


Figura 3.5 Proceso integrante.

Fuente: [12].

3.2.2. Procesos de flujo

Los procesos de flujo principalmente están relacionados en las industrias, una de las aplicaciones más fundamentales es la medición de flujo de agua en estaciones de tratamiento y residencias, se lo puede realizar también a los gases industriales y combustibles[14].

3.2.2.1. Flujo volumétrico

Es la cantidad del volumen real de un fluido que pasa a través de una sección por unidad de tiempo. Sus unidades volumétricas son;

m^3/s , m^3/h , l/h, l/min, GPM (Galones Por Minuto), Nm^3/h (normal metro cubico por hora), etc[14].

$$Q = \frac{V}{t} \quad (3.5)$$

Donde

V= volumen

t= tiempo

Q= flujo volumétrico

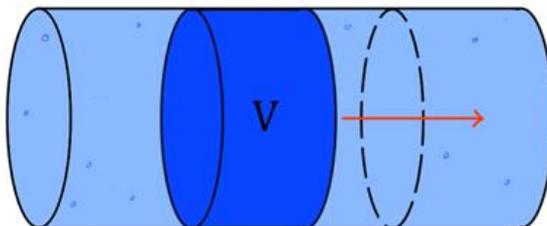


Figura 3.6 Flujo másico de un fluido

Fuente: [14].

3.2.2.2. Flujo másico

Es la cantidad de masa de un fluido que pasa a través de una sección por unidad de tiempo. Sus unidades másicas son: kg/s, kg/h, t/h, lb/h.

$$Qm = \frac{m}{t} \quad (3.6)$$

Dónde:

m= masa

t= tiempo

Qm= flujo másico

3.3. Realidad Virtual

Son imágenes tridimensionales creadas por computadora que permite al usuario interactuar con el entorno utilizando casco o gafas de realidad virtual. La tecnología avanza muy rápido prueba de esto es la realidad virtual lo cual son imágenes que se reproducen cerca de los ojos y da una sensación de inmersión en el entorno, en los videojuegos es donde más se utiliza la realidad virtual, pero se puede aplicar a diferentes campos como la ingeniería para simular procesos y realizar inmersiones en el entorno para interactuar directamente.

Se imaginan volver al pasado o ir al futuro es algo físicamente imposible pero la realidad virtual puede hacerlo posible mediante imágenes tridimensionales muy creíbles que nos permiten viajar a diferentes situaciones como si fuera una máquina del tiempo[16].

3.3.1. Funcionamiento de los dispositivos de Realidad Virtual.

Los dispositivos de realidad virtual son gafas o cascos que recrean imágenes tridimensionales mediante pantallas que se colocan frente a los ojos.

Las pantallas de los cascos o gafas de realidad virtual cerca de los ojos permiten tener todo el campo de visión posible generando una mejor visualización lo cual no se consigue con imágenes proyectadas por otras pantallas.

Se genera una sensación de movimiento adentro del entorno debido a que las gafas o el casco están sujetos a la cabeza, eso quiere decir que los movimientos serán sincronizados tanto en el entorno como físicamente generando que el usuario se sienta parte del ambiente virtual. Las gafas o cascos generan las imágenes mediante pantallas y lentes de visión angular que permite abarcar todo el campo de visión. Para tener una mejor visión se generan imágenes ligeramente diferentes en cada ojo, esto se llama visión estereoscópica haciendo que esas dos imágenes se sientan como una sola imagen tridimensional [17].

En la figura 3.7 se aprecia cómo se recrean las imágenes mediante las gafas de realidad virtual.



Figura 3.7 Como recrear imágenes estereoscópicas en 3D.

Fuente: [17].

3.3.2. Características básicas de la Realidad Virtual

Se pueden distinguir tres fases de la realidad virtual:

Pasivo: Son entornos inmersivos no interactivos esto quiere decir que se puede ver y oír pero muy poco sentir lo que sucede alrededor. El ambiente se puede mover pero no se puede controlar el movimiento.

Exploratorio: Son sistemas que ayudan a la exploración de un ambiente virtual eso quiere decir que solo permite el movimiento son muy usados para observar virtualmente lugares famosos o arquitectónicos[18].

Interactivo: Es el entorno más completo donde se puede explorar e interactuar hasta modificar el entorno virtual.

3.3.3. Tipos de Realidad Virtual

Sistema de sobremesa: Son sistemas en donde se puede desplazar e interactuar por un entorno pero no son inmersivos, las imágenes son proyectadas por un ordenador y no se requieren gafas de realidad virtual.

Sistemas proyectivos: Son sistemas semi inmersivos que intentan generar sensación de inmersión mediante imágenes proyectadas en un lugar cerrado. Muchas de las veces se utilizan gafas de realidad virtual.

Sistemas inmersivos: Son sistemas completamente inmersivos en donde se busca aislar al usuario del entorno real a través de gafas de realidad virtual para conseguir una inmersión completa en el entorno virtual[18].

3.4. Modelamiento 3D

Es el proceso de pasar la representación matemática de cualquier objeto a una representación tridimensional a través de programas especializados, se puede realizar modelados estáticos o que contenga movimiento, o a su vez se puede imprimir en 3D.

3.4.1. Softwares de Diseño CAD

Se dispone de varios softwares para realizar el diseño 3D entre ellos se encuentran los siguientes:

- **Autodesk Inventor:** Es un programa CAD que se utiliza para diseñar piezas mecánicas en 3D, generando un producto final mediante ensambles. Para obtener este producto se debe pagar en la página web oficial de Autodesk pero existe un paquete gratis para estudiantes.
- **Solidworks:** Es un software de diseño CAD 3D para modelar piezas y ensamblajes, permite crear, simular y gestionar los datos del diseño. Para disfrutar de todo el contenido del programa se debe pagar, igualmente existen versiones para estudiantes gratis.
- **CATIA:** Es un programa de informático de diseño, fabricación e ingeniería asistido por computadora, tiene casi las mismas características que Solidwork pero CATIA es mucho más poderoso. Para obtener el producto se debe pagar, existe una versión para estudiantes pero no viene con todas las herramientas.
- **OpenSCAD:** Es un software gratuito diseñado para crear modelados 3D, es un poco complicado manejar este programa porque se debe tener conocimientos de programación.
- **Blender:** Es una multiplataforma gratuita en donde se puede crear modelados, iluminación, renderizar, animación y creación de gráficos tridimensionales.

3.4.2. Diseño mecánico 3D

Una de las ventajas de estos softwares CAD es que se puede diseñar una por una las piezas para después ensamblarles y sacar modelados de productos finales. Una de las herramientas más importantes es que se puede elegir materiales y se puede realizar el estudio de las fuerzas que actúan en cada soporte de los modelados según los pesos de los materiales, las geometrías de las piezas pueden variar con tan solo cambiar las dimensiones haciendo más fácil el modelado 3D[19].

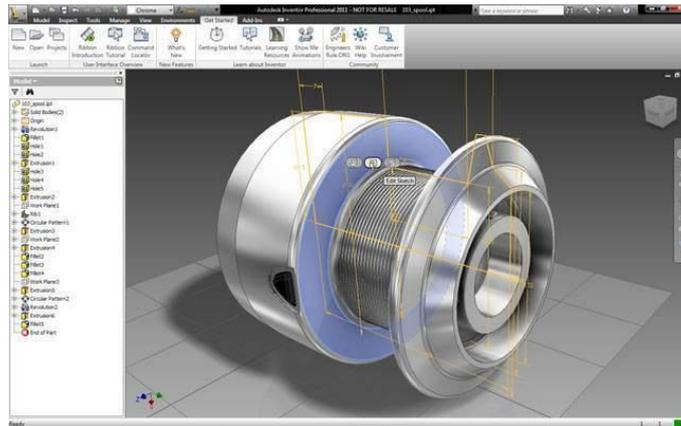


Figura 3.8 Modelado 3D.

Fuente: [19].

3.4.3. Importación de modelamientos 3D a Unity

Para virtualizar una pieza mecánica de ingeniería se lo realiza mediante un software llamado Unity que permite importaciones de modelados 3D desde programas CAD. Unity puede leer directamente archivos “.FBX”, “.dae”, “.3DS”, “.dxf” y “.obj”[17].

Ventajas:

- Solo se exportan los datos que uno necesita.
- Los datos son verificables (se re-importa en el paquete 3D antes que en Unity).
- Generalmente son archivos más pequeños.
- Fomenta un enfoque modular, diferentes componentes para tipos de colisión o interactividad.
- Soporta otros paquetes 3D para cuyo formato Unity no tiene soporte directo.

Desventajas:

- Puede ser un flujo de trabajo más lento para la creación de prototipos e iteraciones.
- Es más fácil perder la pista de las versiones entre la fuente (archivo de trabajo) y los datos del juego (por ejemplo, un FBX exportado).

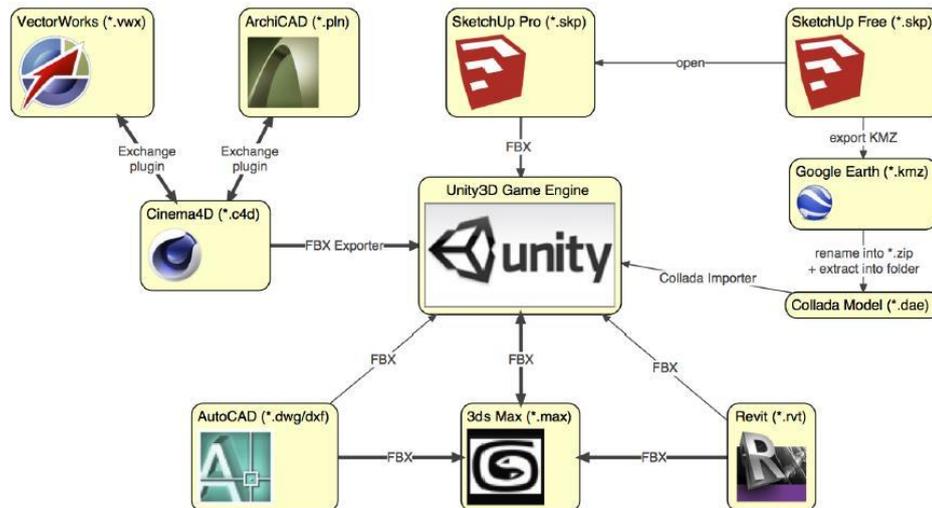


Figura 3.9 Opciones de importación a Unity.

Fuente: [17].

3.5. Softwares de Realidad Virtual

Existen múltiples plataformas para desarrollar la realidad virtual en el mercado unos más costosos que otros, sin embargo se encuentran softwares libres para novatos o estudiantes en donde se pueden crear entornos gráficos para visualizar la realidad virtual.

3.5.1. Concept 3D

Es una plataforma capaz de realizar representaciones 3D con realidad virtual para diferentes actividades como visitas virtuales o mapas interactivos. Se usa más en el sector empresarial para expandir sus creaciones en el mundo virtual. Para obtener el producto con todas sus características se debe pagar, pero Concept 3D tiene una versión gratuita y una prueba gratis.

Características:

- Mapas 3D
- Visitas virtuales 3D
- Integración de datos en vivo
- Compromiso virtual
- Planificación del espacio
- Orientación

3.5.2. Unity 3D

Es un software de desarrollo de videojuegos, la cual tiene múltiples programaciones que permiten diseñar un entorno interactivo. Unity es un programa que permite desarrollar entornos visuales para diversas plataformas y se los realiza mediante editores visuales[20].

Características

- Motor gráfico para renderizar gráficos 2D y 3D.
- Motor físico que simule leyes de la física.
- Animaciones.
- Sonidos.
- Inteligencia artificial.
- Programación o scripting.
- Realidad virtual.

Trasladar modelos 3D diseñados en programas CAD a entornos virtuales que permitan la interacción con la realidad virtual, solía ser muy complicado debido a que se necesitaba un alto conocimiento en programación. El software Unity tiene varias funciones una de ellas es que permite la importación de modelos CAD a los entornos virtuales, reduciendo mucho el proceso de visualizar e interactuar con el contenido en tres dimensiones. Una de las características principales de este software es su gratuidad y accesibilidad por ese motivo se ha escogido para desarrollar el proyecto de tesis.

3.5.2.1. Unity 3D y Realidad Virtual

La realidad virtual de Unity permite tener varios dispositivos de RV sin la necesidad de plugins externos. Cuando se activa la RV en Unity se automatizan varios procesos como la visualización estereoscópica automática, no se requiere tener dos cámaras para la visualización. Las matrices de vista y proyección se ajustan al campo de visión del usuario, las optimizaciones ocurren automáticamente.

3.5.3. Sistemas Inmersivos

Los sistemas inmersivos son aquellos que permiten a los usuarios tener experiencias virtuales visibles, audibles y táctiles muy parecidas a lo que vivimos todos los días en el mundo real. Los sistemas inmersivos buscan ampliar el campo de visión de las personas para generar la sensación de visión envolvente por medio del sistema visual.

Hoy en día hay diferentes dispositivos para tener este tipo de experiencias de inmersión como cascos o gafas que son los principales, además existe otro tipo de sistema el CAVE (Automatic Virtual Environment)[21].

3.5.3.1. Casco de Realidad Virtual, Head-Mounted Display (HMD)

Estos tipos de dispositivos son pantallas que se ubican cerca de los ojos obligando a aislarse del mundo real y obligando a ver la imagen proyectada por el casco de realidad virtual. Este

casco se ubica sobre la cabeza cubriendo la zona de los ojos obligando al usuario a solo ver las imágenes virtuales que se proyectan[21].

Existen 3 tipos de HMD:

- Monocular (una sola pantalla).
- Binocular (dos pantallas que presentan la misma imagen en los dos ojos).
- Estereoscópica (dos pantallas que muestran imágenes independientes).



Figura 3.10 Gafas Oculus Rift de RV.

Fuente: [21].

3.5.3.2. Caverna de inmersión, Cave Automatic Virtual Environment (CAVE)

La caverna de inmersión es un sistema de visualización desde diferentes puntos de vista, cada imagen es proyectada en una pared de un lugar cerrado esto genera sensación de inmersión en un mundo virtual. La Universidad de Illinois construyó en el año 1992 el primer sistema CAVE, que se conformaba de cuatro paredes y un techo como se muestra en la figura 3.11. Hay diferentes implementaciones de los sistemas CAVE a nivel comercial hay modelos de múltiples pantallas. A lo largo de los años se ha venido investigando para mejorar este tipo de sistemas recientemente se presentó un proyecto denominado StarCAVE de tercera generación el cual tiene 15 pantallas distribuidas en forma pentagonal[21].

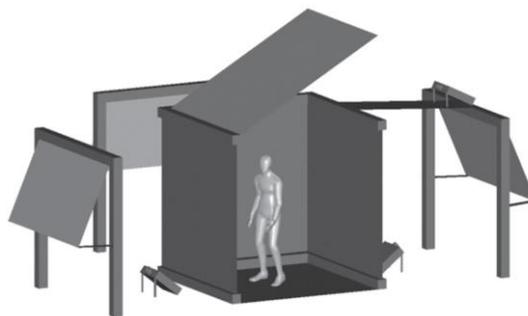


Figura 3.11 Sistema CAVE.

Fuente:[21].

3.5.3.3. Visión estereoscópica

Esta visión estereoscópica es esencial para percibir los elementos más cercanos que otros dan la sensación de profundidad. En pocas palabras es la capacidad que tiene una persona para fusionar dos imágenes que son ligeramente diferentes.

- **Dirección visual y ángulos de convergencia**

La dirección visual es la ubicación que se percibe de un objeto observado, y se genera por el ángulo entre el observador y el objeto. La dirección visual depende del ángulo de la persona para observar un objeto, los ojos del ser humano apuntan directamente a la mitad de las dos imágenes monoculares, para así fusionarlas y ver una sola imagen sin embargo existen estudios que no todas las veces se apunta a la mitad sino a otros puntos entre las dos imágenes. El ángulo de convergencia es un ángulo que forman dos imágenes monoculares está dado por la ecuación:

$$\alpha = 2 \arctan \left(\frac{i}{2D} \right) \quad (3.7)$$

Dónde:

α = es el ángulo de convergencia.

D = es la distancia entre el observador y el objeto observado.

i = es la distancia inter ocular.

En la figura 3.12, se muestra cómo cambia el ángulo de convergencia si comparamos dos objetos ubicados uno más cercano al otro (F_2 y F_1 respectivamente), se observa que el ángulo de convergencia α_2 que se define con respecto a los dos ojos es mayor que α_1 , por lo tanto es posible afirmar que el ángulo de convergencia es inversamente proporcional a la distancia entre el objeto y el observador[21].

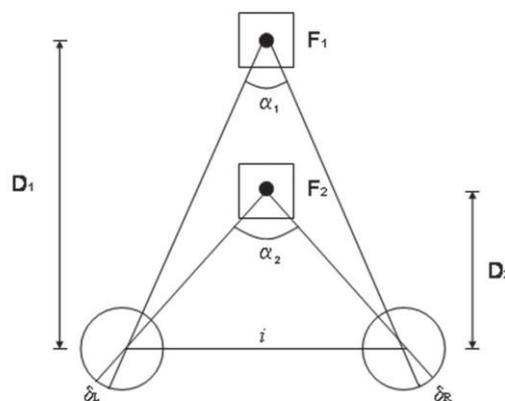


Figura 3.12 Ángulo de convergencia de dos imágenes monoculares.

Fuente:[21].

3.6. Hardware in the loop

Hardware in the loop son sistemas de simulación que consisten en la interacción tanto de componentes reales como componentes virtuales, parte de la simulación está compuesto por un software o una computadora, los mecanismos de control deben utilizar métodos de simulación adecuados, el método de simulación es uno de los más eficientes y precisos en tiempo real para sistemas de control, el hardware in the loop son utilizados para crear entornos de control para posteriormente reemplazar al sistema real por el entorno de simulación por computadora[22].

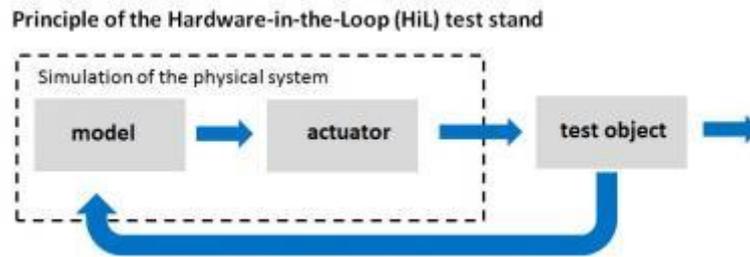


Figura 3.13 Simulación de un sistema Hardware in the loop.

Fuente: [23].

3.6.1. Desafíos y aplicaciones del sistema

Al momento de realizar pruebas en módulos reales tenemos muchos desafíos como;

- Costo de la prueba.
- Costo de la falla.
- Disponibilidad.
- Variación del sistema.
- Repetitividad.

En estas situaciones la simulación en HIL nos da una ventaja de reducción de costos y riesgos[23].

- Mayor confiabilidad.
- Desarrollo más eficiente.
- Menor costo para innovar.

La simulación HIL es utilizada ampliamente en el desarrollo de sistemas de control integrados como:

- Dispositivos médicos.
- Maquinas industriales.
- Sistema de generación de energía.

- Electrodomésticos.
- Aeroespacial.
- Automotriz.
- Control de procesos [23].

3.7. Sistema de Control de Procesos Industriales

Actualmente los sistemas de control de procesos son indispensables en las industrias para mantener una línea de producción estable y sin fallas. Son un conjunto de dispositivos eléctricos, neumáticos, hidráulicos, mecánicos entre otros que están determinados a alcanzar un objetivo. Existen 3 elementos que son la base de un sistema de control:

- Una variable a la que se busca controlar.
- Un actuador.
- Un punto de referencia o set-point.

3.7.1. Control ON OFF

Estos sistemas de control tienen dos estados de operación apagado o prendido, revisa una variable y compara si está por encima o por debajo de un setpoint, como ejemplo tenemos un switch de encendido de una casa. Se podría considerar como un sistema de lazo cerrado pero dependiendo del control también puede ser implementada dentro de un lazo cerrado.

El ejemplo más clásico para entender este sistema es el tanque de agua que tiene un sensor o flotador que abre o cierra las válvulas dependiendo del nivel de agua, cuando el agua está muy por debajo del sensor se abren las válvulas dejando entrar agua al tanque, cuando el agua llega muy por encima del sensor se cierra las válvulas. En la industria aún hay muchos sistemas de control como válvulas hidráulicas las cuales siempre deben quedar cerradas o abiertas nunca a medio nivel, también los relevadores electromecánicos o contactor de alta potencia estos elementos tienen solo dos posiciones on/off[24].

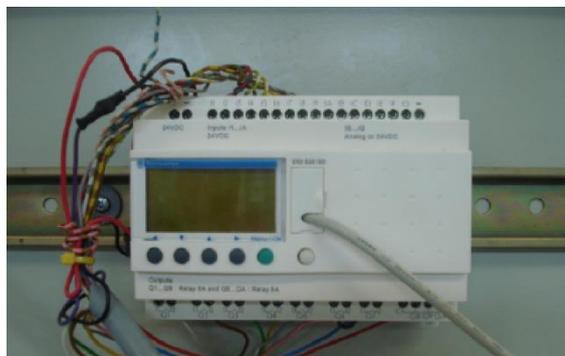


Figura 3.14 Relevador Electromecánico programable.

Fuente:[24].

3.7.2. Control PID

Es un mecanismo de control que mediante un lazo de retroalimentación permite regular variables como la temperatura, presión, velocidad, flujo etc. Calcula la diferencia entre la variable real contra la variable deseada [25].

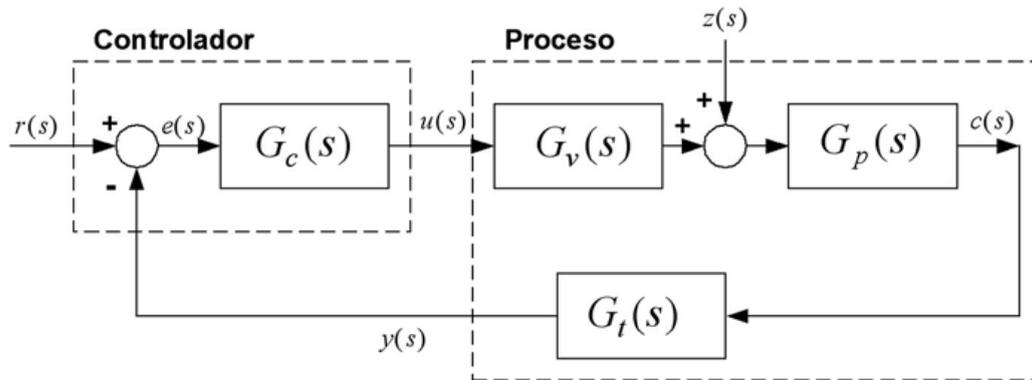


Figura 3.15 Diagrama de bloques.

Fuente: [25].

Las siglas PID significan control Proporcional, Integral y Derivativo, que son las acciones de control aplicadas sobre el error como se muestra en la formula[25].

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int e(t) dt + T_D \frac{de(t)}{dt} \right] \quad (3.8)$$

Dónde:

K_p : ganancia proporcional.

T_i : tiempo integral.

T_d : tiempo derivativo.

3.7.2.1. Estructura del PID

Los miembros de la familia de controladores PID, incluyen tres acciones: proporcional (P), integral (I) y derivativa (D). Estos controladores son los denominados P, I, PI, PD y PID.

P acción de control proporcional: da una salida del controlador que es proporcional al error, es decir: $u(t) = K_P \cdot e(t)$, que descrita desde su función transferencia queda:

$$C_p(s) = K_p \quad (3.9)$$

Donde K_p es una ganancia proporcional ajustable. Un controlador proporcional puede controlar planta estable, pero posee desempeño limitado y error en régimen permanente (off-set).

I acción de control integral: da una salida del controlador que es proporcional al error acumulado, lo que implica que es un modo de controlar lento.

$$u(t) = K_i \int_0^t e(\tau) d\tau \quad C_i(s) = \frac{K_i}{s} \quad (3.10)$$

La señal de control $u(t)$ tiene un valor diferente de cero cuando la señal de error $e(t)$ es cero. Por lo que se concluye que dada una referencia constante, o perturbaciones, el error en régimen permanente es cero.

PI acción de control proporcional-integral: se define mediante.

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(\tau) d\tau \quad (3.11)$$

Donde T_i se denomina tiempo integral y es quien ajusta la acción integral. La función de transferencia resulta:

$$C_{PI}(s) = K_p \left(1 + \frac{1}{T_i s} \right) \quad (3.12)$$

Con un control proporcional, es necesario que exista error para tener una acción de control distinta de cero. Con acción integral, un error pequeño positivo siempre nos dará una acción de control creciente, y si fuera negativa la señal de control será decreciente. Este razonamiento sencillo nos muestra que el error en régimen permanente será siempre cero.

PD: acción de control proporcional-derivativa, se define mediante:

$$u(t) = K_p e(t) + K_p T_d \frac{de(t)}{dt} \quad (3.13)$$

Donde T_d es una constante denominada tiempo derivativo. Esta acción tiene carácter de previsión, lo que hace más rápida la acción de control, aunque tiene la desventaja importante que amplifica las señales de ruido y puede provocar saturación en el actuador. La acción de control derivativa nunca se utiliza por sí sola, debido a que solo es eficaz durante periodos transitorios. La función transferencia de un controlador PD resulta:

$$C_{PD}(s) = K_p + s K_p T_d \quad (3.14)$$

PID: acción de control proporcional-integral-derivativa, esta acción combinada reúne las ventajas de cada una de las tres acciones de control individuales. La ecuación de un controlador con esta acción combinada se obtiene mediante:[26].

$$u(t) = K_p + \frac{K_p}{T_i} \int_0^t e(\tau) d\tau + K_p T_d \frac{de(t)}{dt} \quad (3.15)$$

Y su función de transferencia resulta

$$C_{PID}(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (3.16)$$

3.7.3. Control Moderno

Este tipo de control ha existido desde hace mucho tiempo ahora la mayoría de las industrias tienen sistemas de control automáticos modernos para su línea de productos. Hoy en día es muy difícil encontrar industrias que no contengan algún sistema de control retroalimentado. Ejemplos de controles modernos tenemos una fábrica de autos donde tienen una línea de ensamblajes previamente programados. Una central nuclear es otro buen ejemplo donde la temperatura del reactor está monitoreada en tiempo real y el agua para enfriar el reactor se controla mediante sistemas de controles modernos automáticos. Este control está ligado a la robótica en donde deben existir cero tolerancias a los errores porque se afectaría productos en línea de ensamblajes[24].



Figura 3.16 Línea automatizada de ensamblajes de autos.

Fuente: [24].

3.7.4. Control Fuzzy

El control Fuzzy se expresa con un control en base a palabras en vez de los números, todas las variables como temperatura, flujo o nivel se miden básicamente en números, estos se tienen que realizar cambios o adaptaciones antes de la colocación de la variable al controlador a todo esto se lo llama fusificación [27].

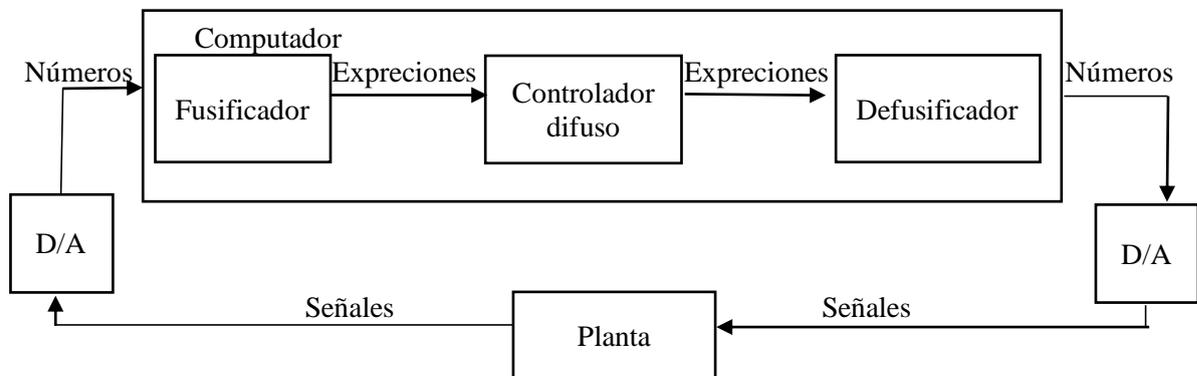


Figura 3.17 Lazo de control difuso.

Fuente: [27]

La etapa de fusificación suministra grados de longitud diferentes para los datos de entradas, para todo esto se busca relación entre el estado de la variable y la función de pertenencia, desde luego que algunas funciones de pertenencia serán mayores a otras expresiones debido a los valores de la variable. Cuando se haya determinado el lenguaje del estado de las variables recién ahí se podrá disponer de una relación lógica entre ellas que aclaran el sentido común y permiten producir las operaciones requeridas que se deberían aplicar en los procesos. Todos estos datos se deben convertir a digital que sería la defusificación esto se logra mediante un convertidor para que la señal se pueda utilizar en la industria[27].

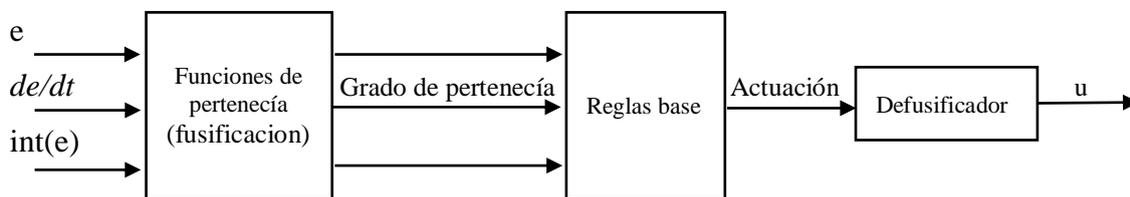


Figura 3.18 Diagrama de control Fuzzy.

Fuente: [27].

3.7.5. Control Predictivo

El control predictivo es uno de los más actuales y no tiene una estrategia de control específica, sino que cuenta con varios conjuntos de métodos que utilizan para recolectar señales de control. Generalmente estos modelos se usan para predecir la señal de salida en función de las señales de entrada conocida.

Todas las medidas en los procesos de control más actuales se calculan de manera diferente al resto mediante un programa de optimización en donde se toma en cuenta las posibles limitaciones, por ende el proceso es el que resalta en el controlador. Los modelos seleccionados deben tener claro la dinámica de los procesos para poder predecir con la mayor precisión posible toda la evolución del sistema, al mismo tiempo debe ser simple de implementar y de comprender. Todos los métodos de control predictivo son relativamente diferentes al tipo de modelo que se va a utilizar. Cuando la función de costo es cuadrática y el modelo es lineal se podrá tener una solución clara de todo el proceso si este no sería el caso se tendría que utilizar algoritmos complejos de optimización numérica. Los resultados del problema de optimización dependerán de varios aspectos como los números de variables, el control y predicción de números[28] .

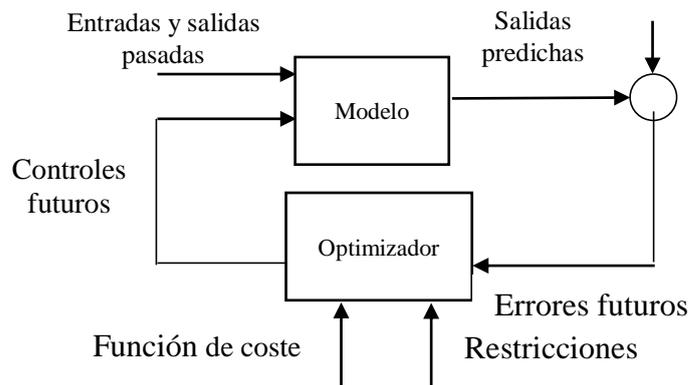


Figura 3.19 Estructura de un control predictivo.

Fuente: [28].

4. METODOLOGÍA

El presente proyecto de tesis se orienta hacia un estudio de tipo descriptivo. Cada una de las etapas presenta una relación clara y concisa para el desarrollo general. La base fundamental del proceso se encuentra dirigida hacia la recopilación de la información y la inclusión de todas las temáticas de control, automatización y la descripción de componentes.

El plan de tesis se apuntaló en tres diferentes propiedades, una investigación de campo, de manera que se pudo analizar las variables de investigación por medio de la observación y manipulación de un equipo práctico expuesto en los laboratorios de la Universidad, las mismas que serán manejadas dentro de un algoritmo de control que conlleva a una emulación más realista, la investigación bibliográfica, permite iniciar la búsqueda de antecedentes e investigaciones de relevancia en textos impresos como digitales con respecto al diseño y simulación del proceso de control de flujo, desarrollando un boceto CAD en 3D, para luego trasladarlo a un motor gráfico (Unity 3D) y obtener la emulación de este tipo de proyecto, además permite establecer el correspondiente marco teórico y una correcta fundamentación científica, la investigación experimental, ya que se realizó varias pruebas de control de las variables de la variable flujo.

Para el proceso de la investigación científica, la técnica es una herramienta de fundamental importancia, ya que de ella dependerá la organización de la investigación.

Una técnica investigativa puede tener los siguientes objetivos:

- Aportar instrumentos para manejar la información.

- Ordenar las etapas de la investigación.
- Llevar un control de los datos.
- Orientar la obtención de conocimientos.

En el presente proyecto de tesis se utilizó las siguientes técnicas:

- Observación. - Comportamiento de instrumentos análogos.
- Medición. - Toma de datos de valores reales de sensores de nivel y flujo.
- Modelación. - Diseño CAD de un diagrama P&ID del banco de pruebas.
- Simulación. – Entorno virtual de proceso control de flujo.

El presente proyecto utiliza el método cuantitativo ya que influyen principalmente las variables de flujo y nivel las cuales se van a simular mediante un algoritmo programable dentro del ambiente y a la vez permitirá al usuario la manipulación de las mismas.

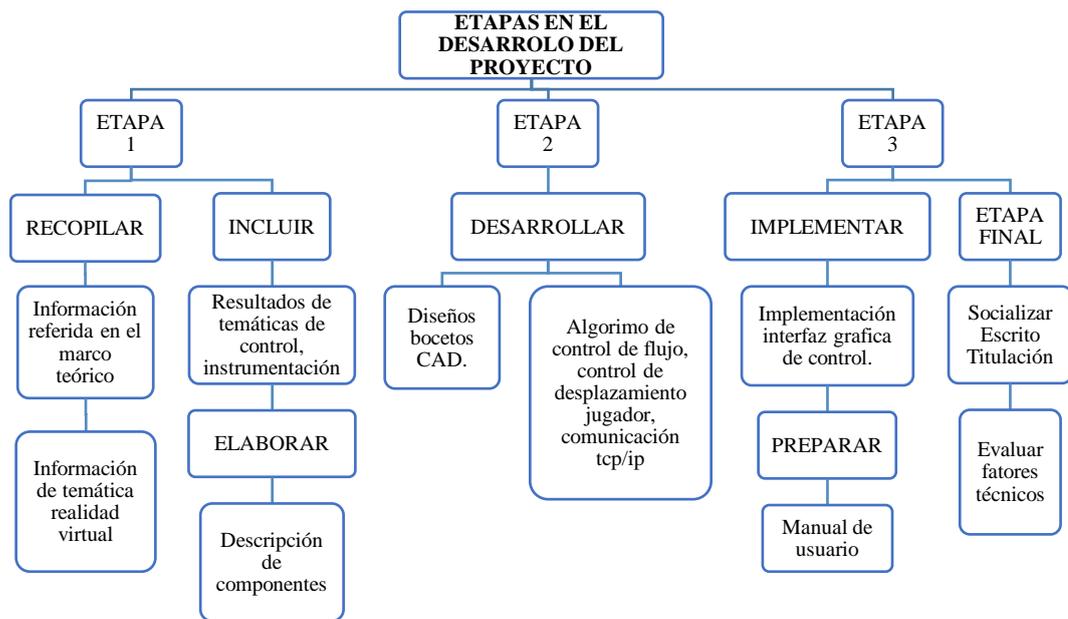


Figura 4.1 Desarrollo del proyecto.

Este capítulo detallara la metodología para la implementación del control de flujo y el entorno virtual para el aprendizaje del control de procesos.

El esquema que describe el funcionamiento del sistema didáctico se muestra en la figura 4.2.



Figura 4.2 Sistema Didáctico.

El sistema requiere la interacción entre LabVIEW y Unity 3D para el intercambio de los datos relevantes en el proceso a controlar, para lo cual, en primera instancia, se han desarrollado las programaciones y diseño de interfaces en cada uno de forma separada, para después implementar un método conexión entre sí a través del protocolo de comunicación tcp/ip. Además, se hace uso de dispositivos periféricos como Oculus Rift los cuales proveen la inmersión e interacción requeridas en el entorno.

4.1.Implementación del control

4.1.1. Diagrama P&ID de la estación de control de flujo

En la figura 4.3 se muestra el diagrama P&ID de la estación de control de flujo en la que se implementará la estrategia de control de flujo, mismo que será utilizado como base para el diseño del entorno virtual.

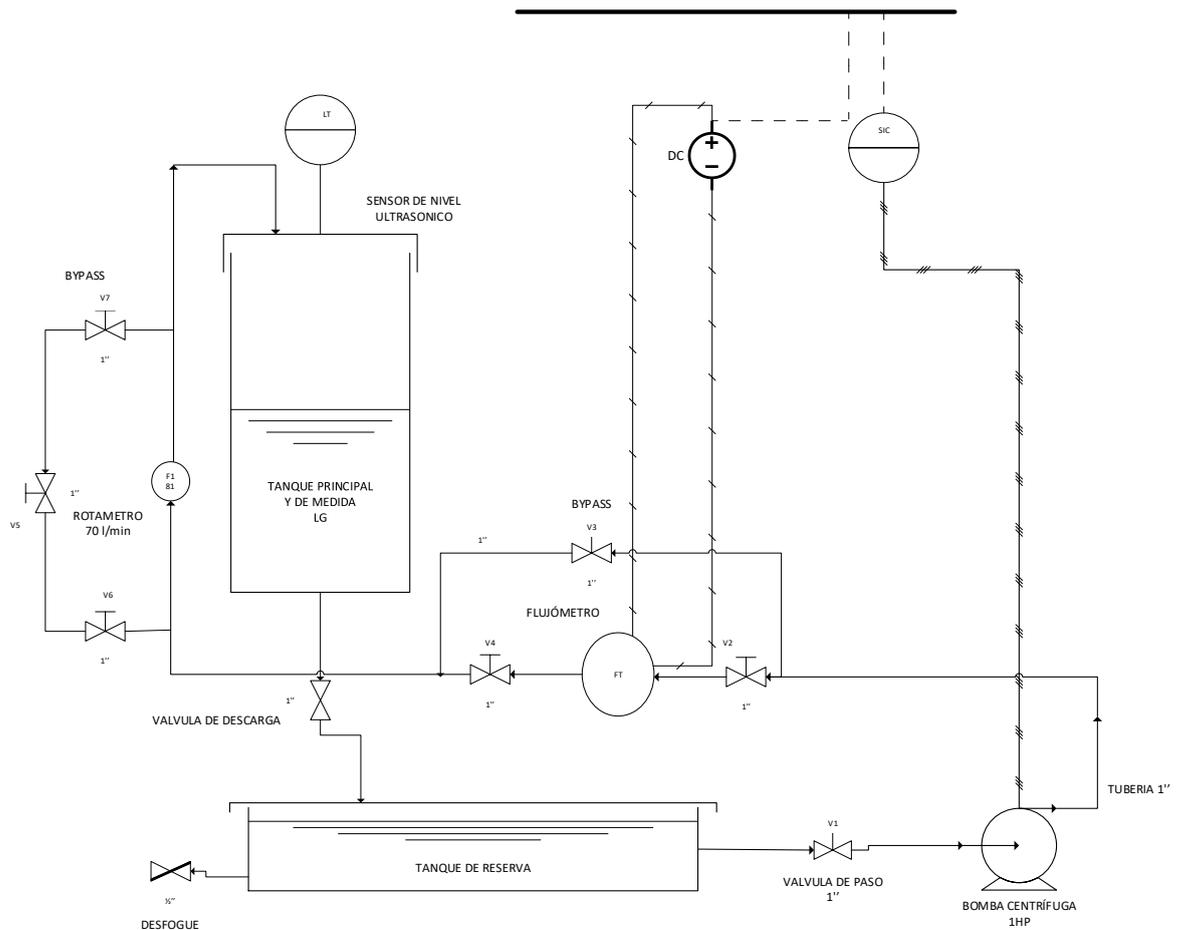


Figura 4.3 Diagrama P&ID.

En el diagrama mostrado en la Figura 4.3 se observa un banco de pruebas de un circuito cerrado de circulación de un fluido, que está almacenado en un tanque reservorio que será impulsado por un bomba centrífuga hacia un tanque elevado, el circuito está formado por un tramo de tubería donde se encuentra ubicado dos sensores de caudal un digital y un rotámetro de boya que nos indicará los valores de flujo a través de las tuberías, luego de almacenar, el fluido retorna hacia el tanque de depósito por un tramo de tubería conectada al tanque de reserva que está controlada por válvula de vaciado.

El sistema presenta dos de las variables a controlar que son el flujo y el nivel de tanque superior que en el caso de este proyecto nos enfocaremos en control de flujo.

4.2. Determinación de la función de transferencia.

Para determinar la función de transferencia se define la ecuación de estado del sistema que representa un punto de equilibrio donde el caudal de entrada es el resultado de caudal de las salidas más la variación de caudal volumétrico.

$$Q_{in} = Q_{out} + \frac{dV}{dt} \quad (4.1)$$

Considerando que la variable de manipulación es el caudal de entrada y a su vez la variable del proceso, como también se deduce como la variable de concentración en el tanque elevado. Se define una ecuación diferencial.

$$V \frac{dy}{dt} = Q(U - y) \quad (4.2)$$

Donde aplicado la transformada de Laplace se obtiene que la función de transferencia de flujo se define como:

$$G(s) = \frac{1}{s\left(\frac{V}{Q}\right)+1} = \frac{1}{sT+1} \quad (4.3)$$

Donde la $T = \frac{V}{q}$ es constante de tiempo.

Considerando que el flujo de trascorrir por medio de una tubería hacia el tanque elevado una distancia considerable se emplea una función de retardo.

$$G(s) = e^{-sL} \quad (4.4)$$

Motivo por el cual se modifica nuestra ecuación de transferencia para el sistema indicado anteriormente de la siguiente forma.

$$G(s) = \frac{1}{sT+1} e^{-sL} \quad (4.5)$$

Para hallar los valores de constantes T y L se genera una señal de entrada a distintas velocidades que serán ingresados en función de ident de Matlab que nos arrojará una función de transferencia.

$$G(s) = \frac{0.99283}{0.5854s+1} e^{-1.316s} \quad (4.6)$$

La obtención de la función de transferencia se la obtuvo de un módulo físico del laboratorio de instrumentación de la carrera de Ingeniería Electromecánica, en la planta tenemos que la entrada del sistema es el variador de frecuencia el cual permite manipular la señal de escala de 0 a 10 voltios, manipulamos la variable en voltios y obtenemos como resultado una influencia en el flujo el cual va a disminuir o incrementar, el voltaje que se le aplique se transforma en flujo. En el anexo 3 se visualiza que la recolección de datos del módulo real, para la obtención se lo realiza mediante una tarjeta de adquisición de datos.

Las señales de flujo obtenidas son bastantes turbulentas y para observar de mejor manera sin muchos ruidos se utiliza un filtro en las frecuencias. Para la obtención de la función de transferencia utilizamos el modelo de escalón donde podemos ir dando valores los cuales podemos observar cómo va reaccionando el sistema.

Para los primeros datos se realizó con el primer escalón de 30 voltios el cual utiliza un control de lazo abierto donde se obtiene las ecuaciones para transformar el voltaje en caudal,

en este caso utilizamos dos ecuaciones una escala común que son los litros/ minuto, por un lado tengo el sensor de 1 a 5 voltios y del otro lado tengo el variador de 0 a 10 voltios y tenemos una escala en común que es el caudal.

Para la obtención de estos datos se realizó la siguiente prueba dando voltios al variador observamos cuantos voltios obtengo en el sensor y cuanto de caudal, en la planta real al dar el valor de 1 a 2.2 v se puede apreciar que la bomba empieza a girar pero la velocidad demasiada baja para que el agua pueda ser impulsada y medida, en 2.2 v el agua comienza a circular y se puede observar el chorro pero sin embargo ya hay un inconveniente con el sensor no me detecta debido que es muy bajo y el sensor sigue en cero, la zona de 2.3 a 2.4 v en el variador el sensor es bastante inestable ya comienza a marcar valores pero son erróneos de no marcar nada empieza a marcar valores erróneos, de los 3 v en adelante hasta cuando le damos 10 v el sistema nos comienza a marcar datos en el sensor, al momento de darle 10 v es el límite de la estación de caudal el sensor nos marcará el caudal máximo.

Con la función de transferencia realizamos la simulación de la señal de flujo en LabVIEW



Figura 4.4 Control PID en LabVIEW.

4.3. Diseño simulación en LabVIEW

Para el desarrollo de la simulación del módulo de control de flujo se utiliza la plataforma de LabVIEW donde se diseña un diagrama esquemático con todos los componentes y variables que forman parte del sistema de control de flujo.

En la figura 4.5 se observa los diferentes componentes como tanques, válvulas, bomba, sensores

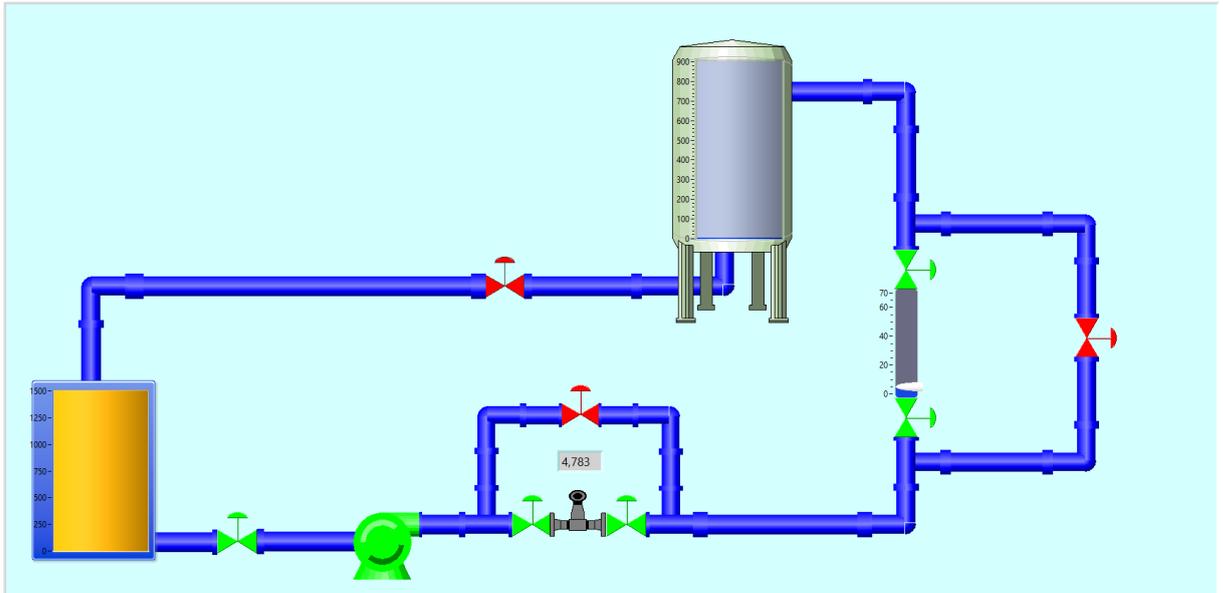


Figura 4.5 Diagrama esquemático.

Además se puede diferenciar las variables manipulables (set point, constantes de ganancia PID, función de transferencia, rango de apertura de válvulas entre otras) y variables de inspección (respuesta de PID, variable de proceso, nivel de los tanques etc).

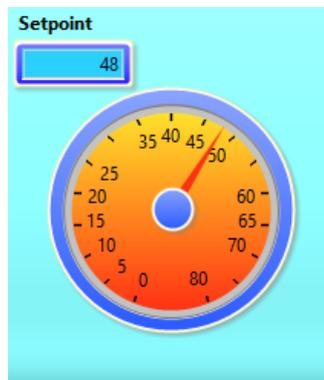


Figura 4.6 Setpoint.

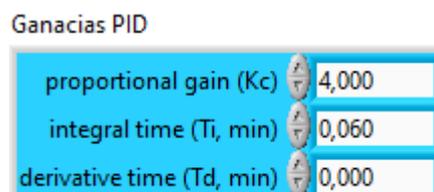


Figura 4.7 Ganancias PID.

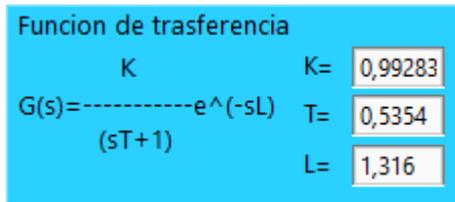


Figura 4.8 Función de transferencia.

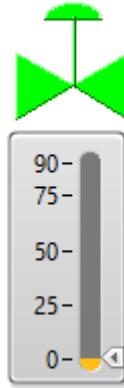


Figura 4.9 Variación de válvula.

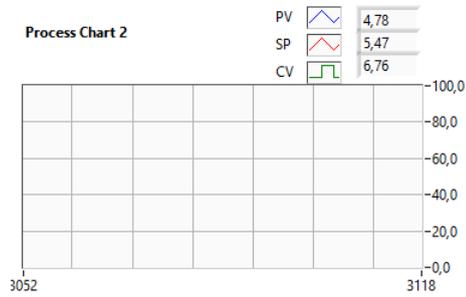


Figura 4.10 Respuesta del PID

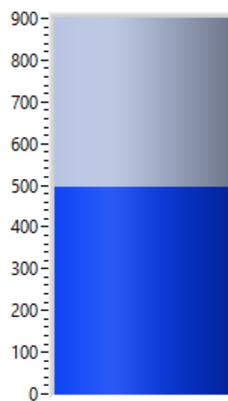


Figura 4.11 Nivel de tanque.

Concluido el diagrama de interfaz (HMI) se procede al desarrollo de la programación en el cual primero se crea un bucle tipo while, añadiendo un shift register ver figura 4.12 el cual permitirá leer y registrar los valores de la variable de flujo y/o proceso.

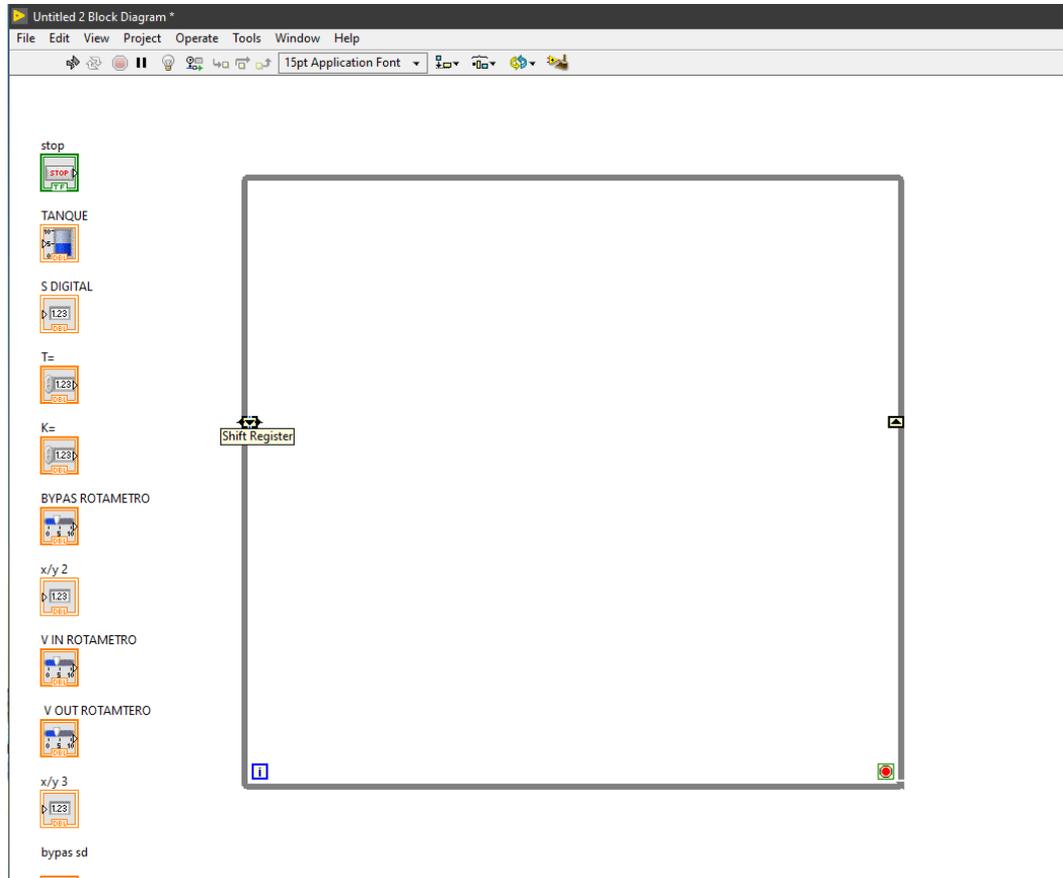


Figura 4.12 Estructura While.

Al generar la simulación de la variable de proceso se debe diseñar un componente que represente la planta de proceso, en LabVIEW existe un infinidad de poder reproducir una variable fluctuante en el tiempo que en el caso particular de este proyecto se utiliza plant simulation.vi (ver figura 4.13) que es un VI desarrollado por la plataforma National Instruments que representa específicamente un planta para el control de fluidos[29].

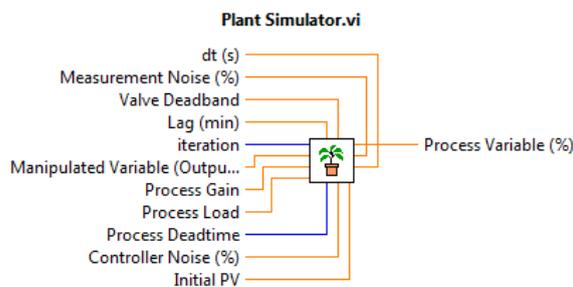


Figura 4.13 Plant simulation.

Fuente: [29]

Plant simulation.vi está regido por una función de transferencia similar a $G(s) = \frac{K}{sT+1} e^{-sL}$, donde permite ingresar sus constantes en las siguientes entradas donde la constante de ganancia K está definida por Process Gain, T definida en Process Deadtime y L en la entrada Lag (min) como se muestra en la figura 4.14.

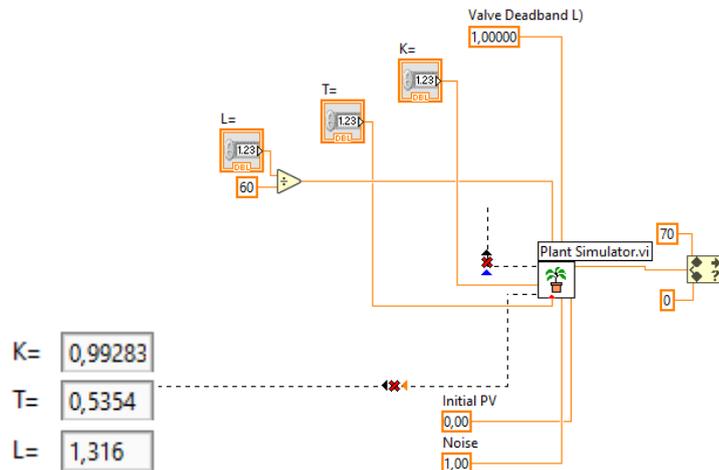


Figura 4.14 Constantes.

Cuyo valor debe estar definido en minutos, una razón por la escogencia del Plant simulation.vi es que además se puede cargar valores de ruido y un valor inicial en la variable del proceso, como se observa en la figura 4.15 se puede añadir carga al sistema en la entrada de Process Load lo que nos permitirá simular perturbaciones que estará determinadas con el rango de apertura de las válvulas.

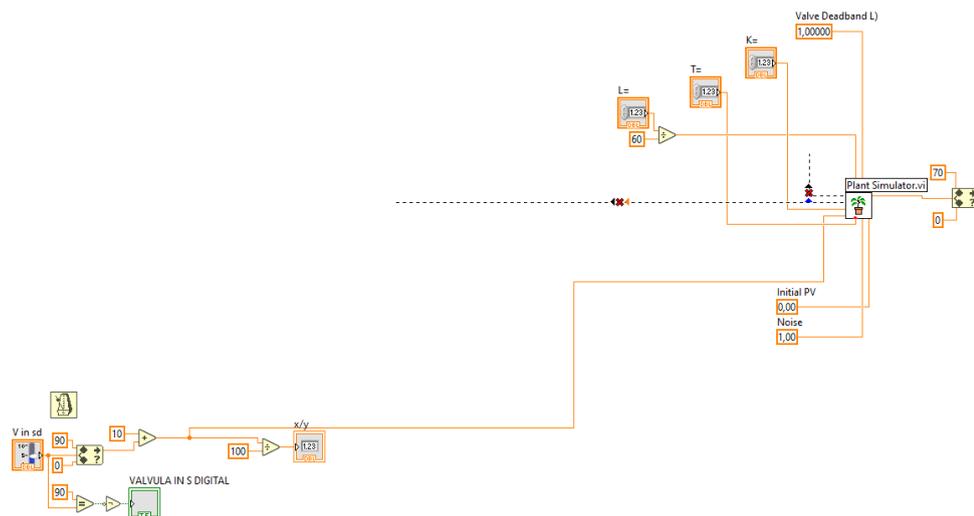


Figura 4.15 Process Load.

Para poder controlar el valor de flujo a un valor seteado por el usuario se añade un controlador PID, como se muestra en la figura 4.16.

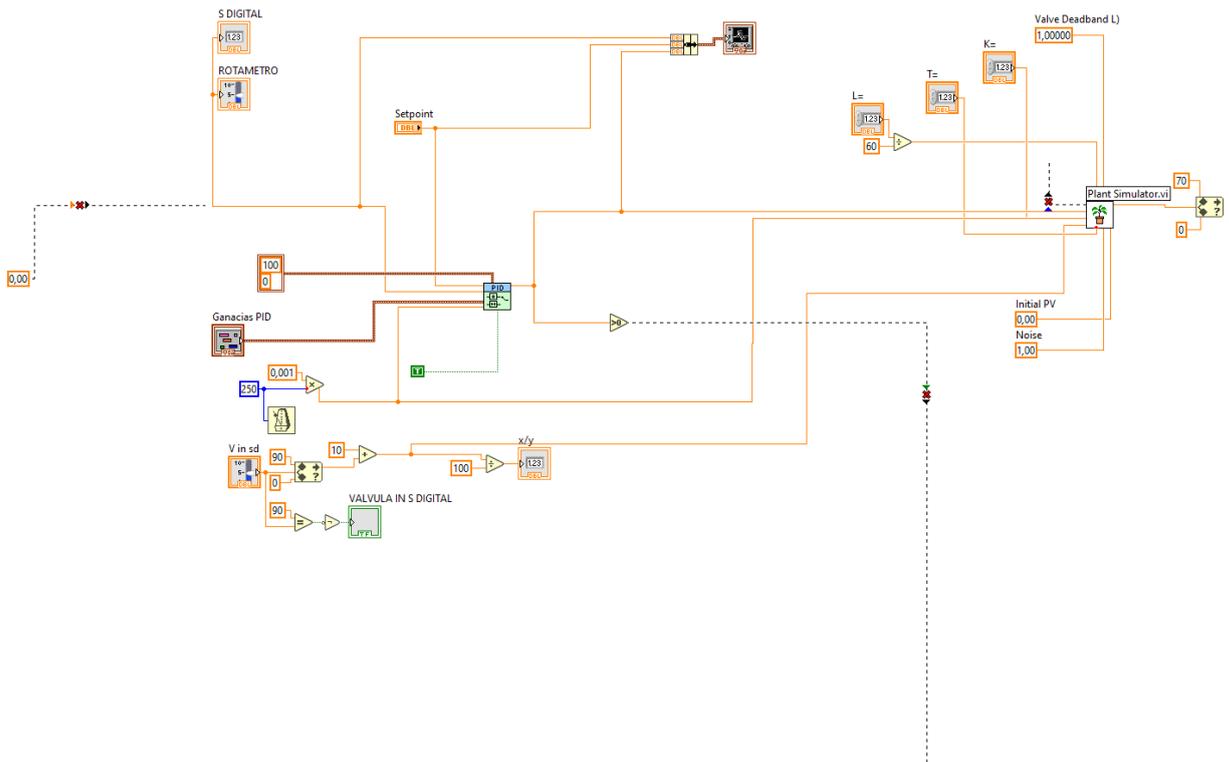


Figura 4.16 Controlador PID.

Como se observa en la figura el control PID permitirá controlar la variable porcentual de la velocidad de la bomba que aplicada a la planta simulada se podrá controlar en valor del flujo a través de la tubería en un rango de 0 a 80 Lts/min, para sintonizar las constantes de ganancia del controlador PID se utiliza método lambda, ya que en la función de transferencia presenta un tiempo muerto definido por la constante e^{-sL} obtenido como resultado los siguientes valores.

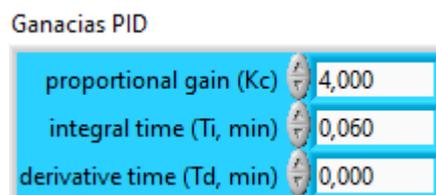


Figura 4.17 Valores de ganancias PID.

Para simular el llenado y vaciado de los tanques tanto como el elevado y el de depósito se utiliza un segundo bucle que estará controlado por un tiempo variable en función del valor mostrado en la lectura del sensor digital que hará que varíe la velocidad del llenado o vaciado de los tanque creando el efecto de variación de flujo.

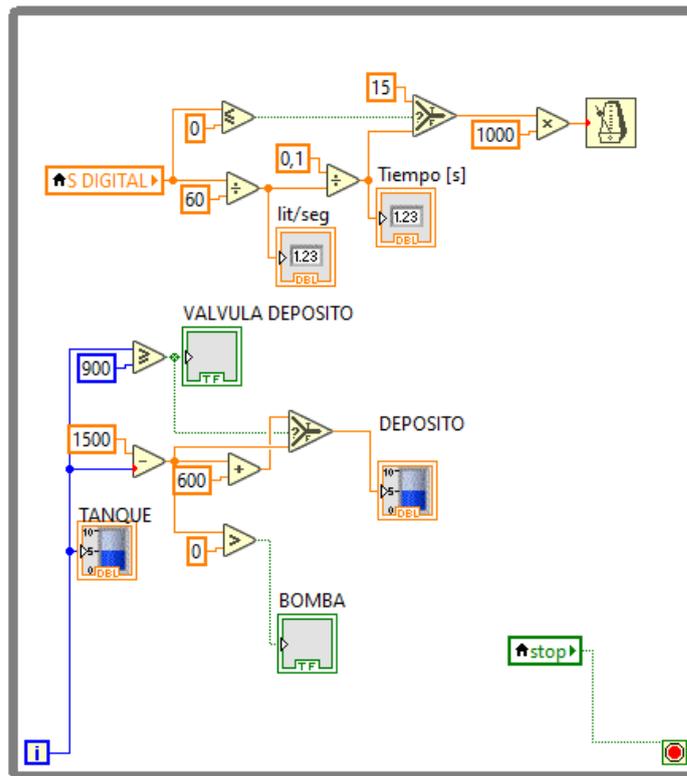


Figura 4.18 Llenado y vaciado de tanques.

4.4. Diseño de la planta en Inventor 2020

Inventor es un programa de diseño, con un gran conjunto de herramientas para modelar elementos de CAD 3D por lo cual es seleccionado con un preprocesador para diseño de planta ya nos ofrece facilidades de uso para lograr el modelo deseado. Posee modelos básicos para representar válvulas, tanques, bombas y accesorios de tuberías comunes en industrias. Para esto cuenta con un amplio catálogo de tuberías capaz de satisfacer las necesidades básicas de un diseño industrial casi en su totalidad.

El primer paso es trazar los bocetos en 3D de los tanques, bombas y soporte tal como se muestra en la figura 4.19.

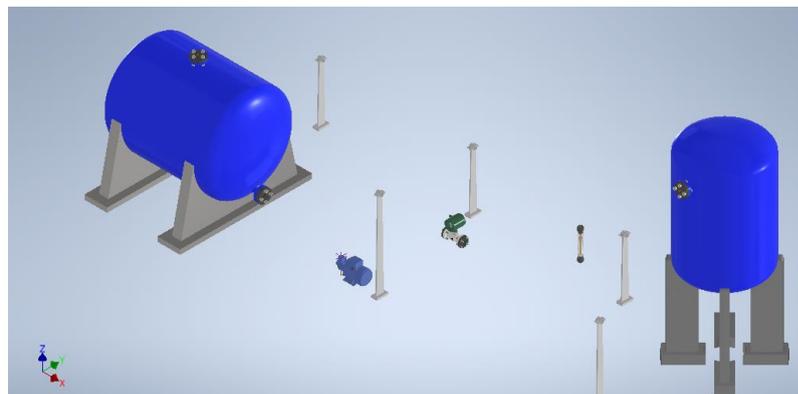


Figura 4.19 Soportes, tanques y bombas.

El ruteo de tuberías es una de las principales características por la cual se ha utilizado esta herramienta. Puede hacerse a partir de un punto de inicio o nodo abierto hasta un punto final, teniendo como referencia por defecto los puntos del eje central de la tubería, además, ofrece opciones de ruteo dependiendo de los objetos existentes en el modelo y la orientación de los puntos de conexión, tal como se muestra en la figura 4.20.

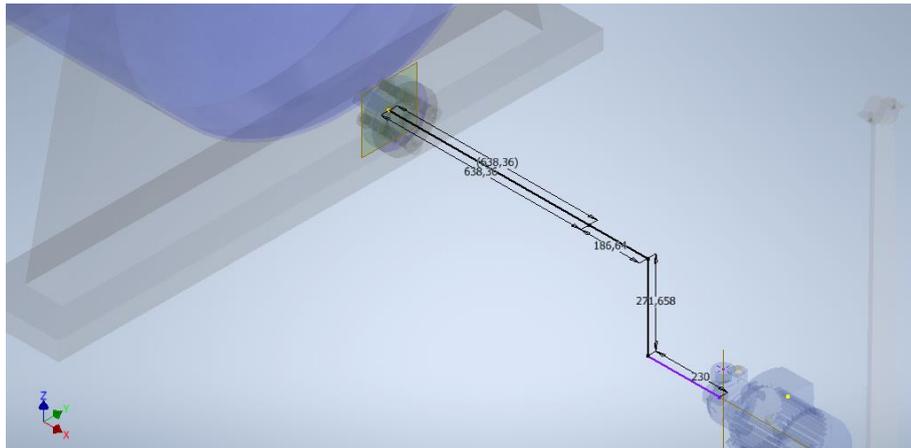


Figura 4.20 Ruteo de tuberías.

Una vez determinada la ruta del conducto aplicamos el llenado de enrutamiento para que el programa genere la tubería de acuerdo a la norma seleccionada. ASTM D 1785-ASTM D 2466 - Tubería articulada de PVC y accesorios como se muestra en la figura 4.21 y 4.22.

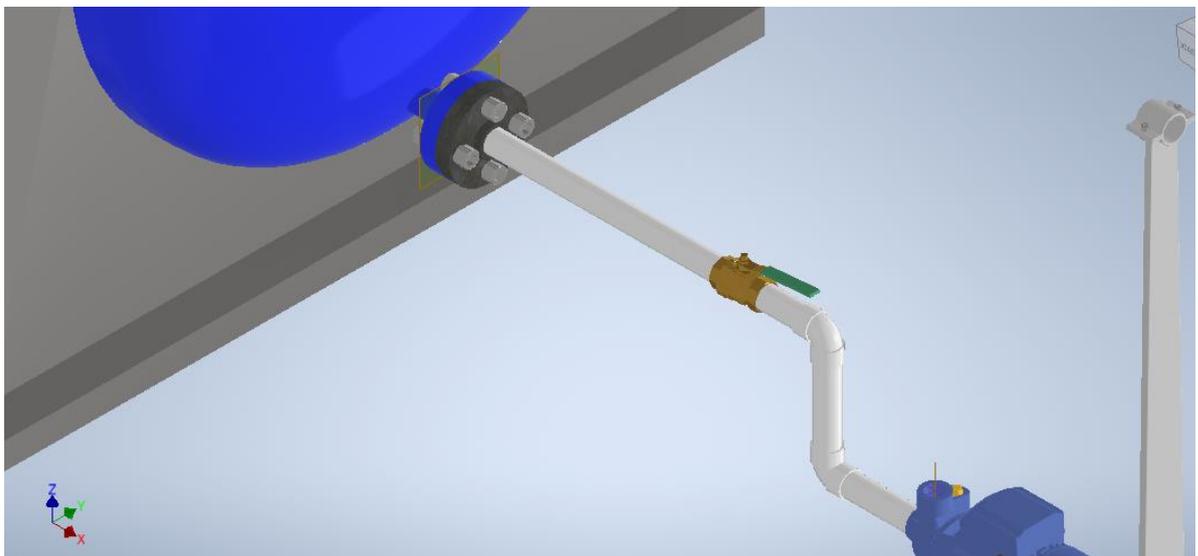


Figura 4.21 Tubería completada.

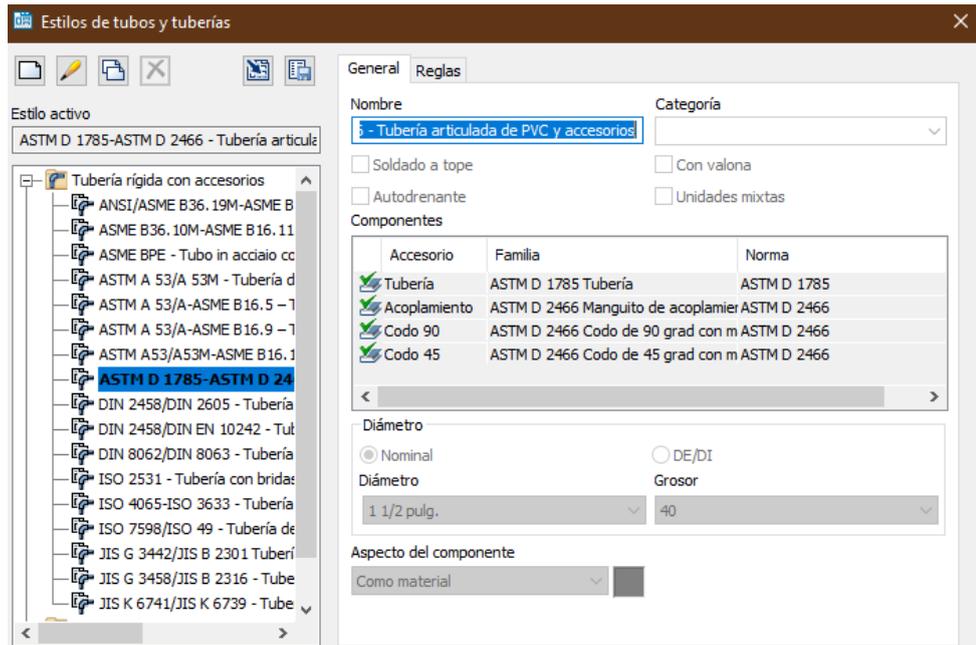


Figura 4.22 Norma seleccionada para tubería.

Aplicamos los siguientes enrutamientos para lograr el diseño final de la planta 3D como en la figura 4.23.

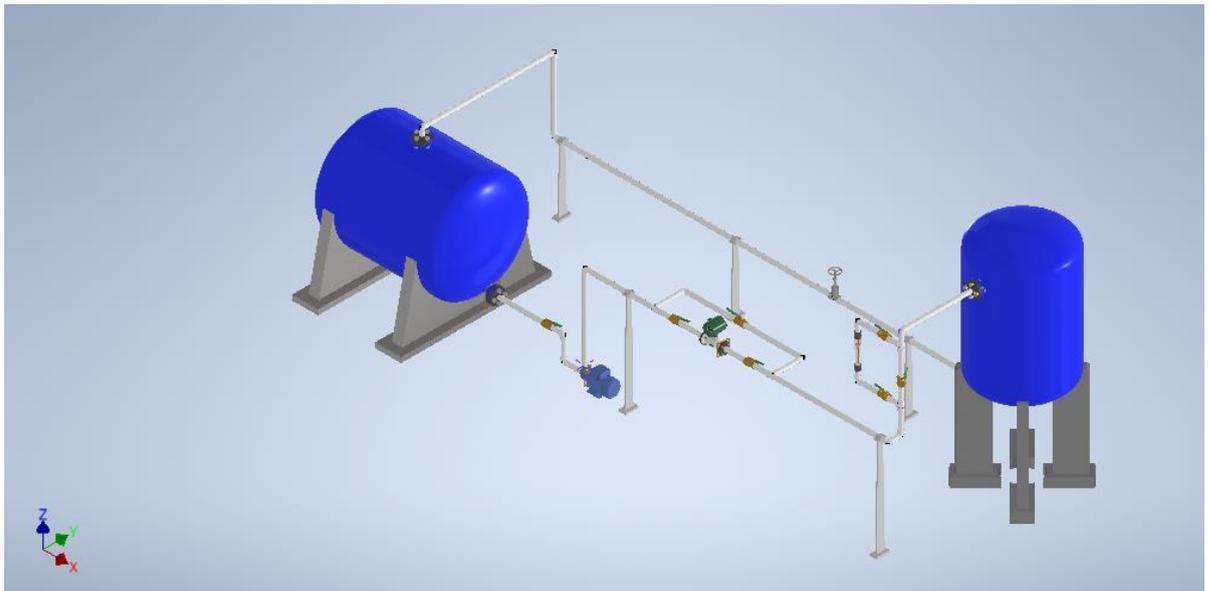


Figura 4.23 Diseño final de la planta.

4.5. Diseño del ambiente virtual en Unity 3D

El modelo 3D diseñado en Inventor se importa en Unity 3D con el formato *.obj, donde se deben añadir texturas a todos los objetos para obtener un entorno virtual realista. Además, se añadieron varios objetos que contribuyen al realismo y comodidad para el usuario, pero no interfieren en la funcionalidad del entorno. El entorno virtual con todas sus texturas se muestra en la figura 4.24.

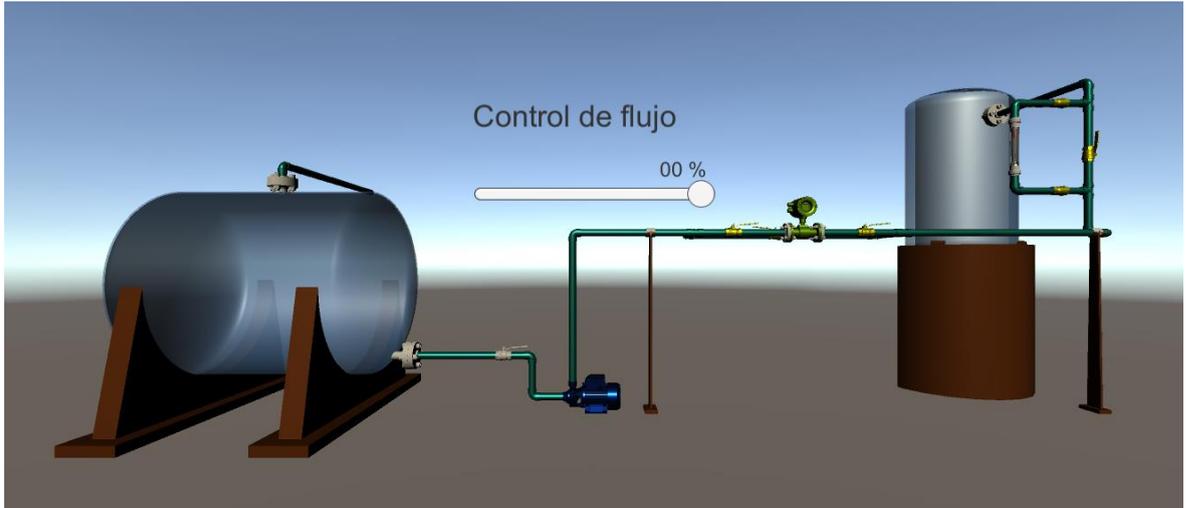


Figura 4.24 Entorno virtual en Unity 3D.

Para obtener el control del proceso se crea un HMI, utilizando objetos de Canvas que son objetos bidimensionales como botones, sliders, textos, etc. Visualizando 2 pantallas que contienen toda la información relevante del proceso como el nivel del tanque que estará controlado por la cantidad de flujo en la tubería que se encuentra comandado por la ecuación, que representa la cantidad de fluido a través de una válvula, alarmas que será activadas o desactivada cuando presente alguna alteración en el correcto funcionamiento de proceso, entre otros. También se crearon gráficas de tendencias para visualizar las variables de proceso y de control. Las pantallas con las que cuenta el HMI se muestran en la figura 4.25.



Figura 4.25 Pantalla HMI.

El siguiente paso es crear un objeto que funcione como jugador en primera persona. Para esto se añade un objeto vacío el cual debe contar con un script que permita su movimiento, además de contener al objeto “MainCamera” (objeto que contiene el componente Camera

para visualización de modelos 3D y el componente AudioListener para escuchar sonidos en el entorno) que servirá como medio único de visualización. El objeto, con el nombre “Player”, que cuenta con estos atributos para poder desplazarse a través del entorno se muestra en la figura 4.26.

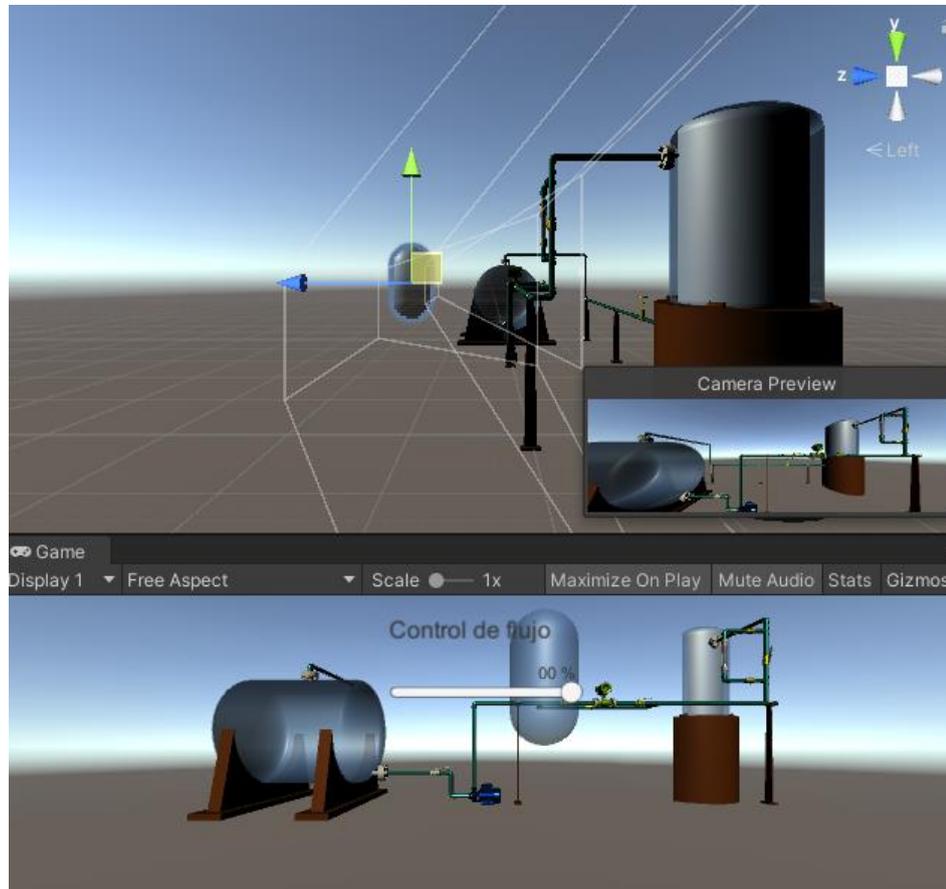


Figura 4.26 Visualización de objeto en primera persona.

4.6. Integración del dispositivo Oculus Rift en Unity 3D

Oculus Rift es un dispositivo para realidad virtual capaz de proveer una alta inmersión dentro de entornos virtuales este objeto posee un sistema de giroscopio que permite controlar los movimientos de la cabeza del usuario para cambiar su posición en el ambiente virtual, lo cual ofrece una impresión de presencia en dicho ambiente. El módulo Oculus Rift se muestra en la figura 4.27.



Figura 4.27 Gafas Oculus Rift.

Fuente: [21]

Para poder utilizar Oculus Rift es necesario descargar e instalar la aplicación para la gestión y configuración del dispositivo desde su web oficial. Dicha aplicación se muestra en la figura 4.28.

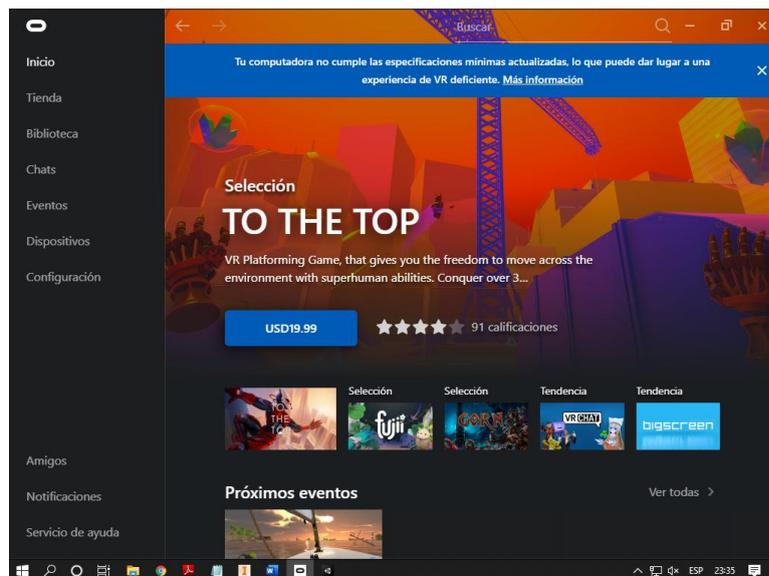


Figura 4.28 Aplicación de Oculus.

Para poder utilizar Oculus Rift en Unity3D se debe habilitar la opción “Orígenes desconocidos” en el menú Settings>General como se muestra en la figura 4.29.

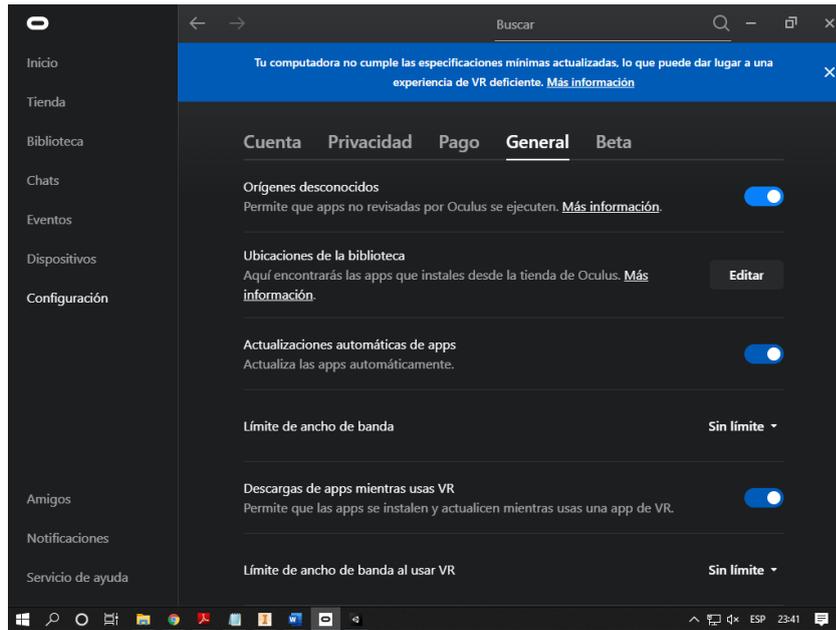


Figura 4.29 Habilitar Oculus Rift.

Al habilitar esta opción Oculus Rift puede empezar a funcionar en aplicaciones externas además se debe instalar un paquete en Unity “Oculus Integration” que tiene una serie de assest que mejora la capacidad de Oculus Rift de inmersión en el entorno de Unity 3D.

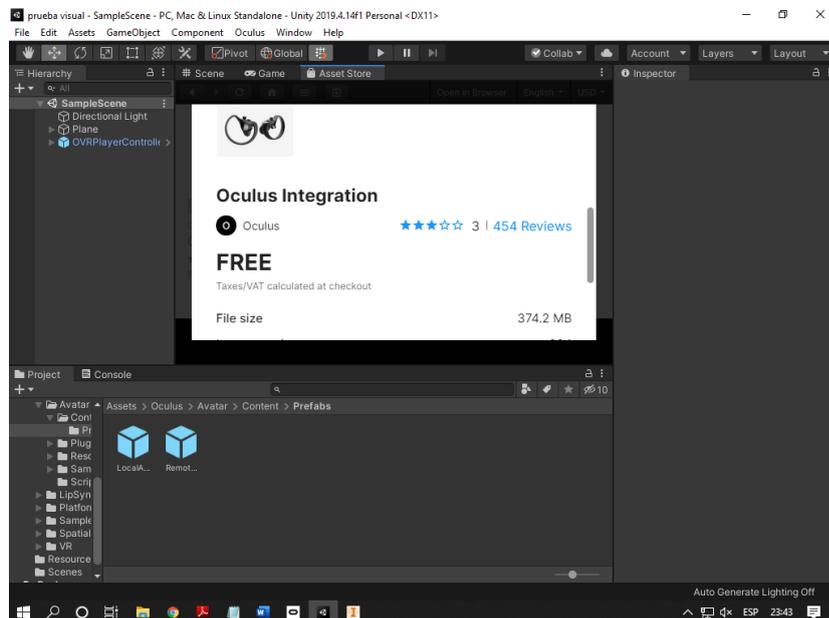


Figura 4.30 Paquete de Oculus en Unity 3D.

La orientación del usuario dentro del entorno estará determinada por la rotación del objeto “MainCamera”, mismo que rota según la orientación de Oculus Rift como se muestra en la figura 4.31, una vez habilitado el soporte para realidad virtual. Con esto y la programación desarrollada para el movimiento del player, el usuario puede desplazarse libremente a través del entorno virtual en su totalidad.

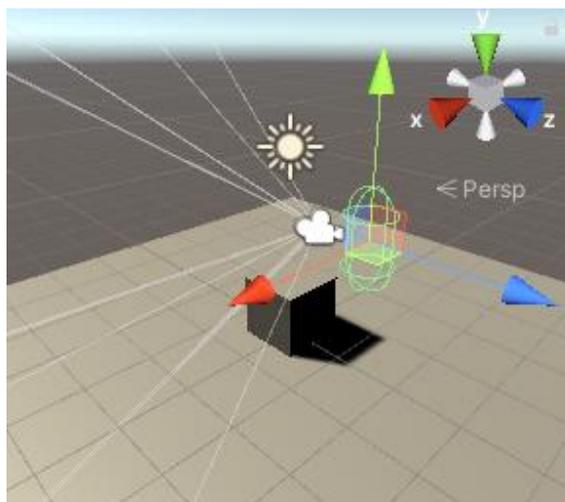


Figura 4.31 Tipo de orientación.

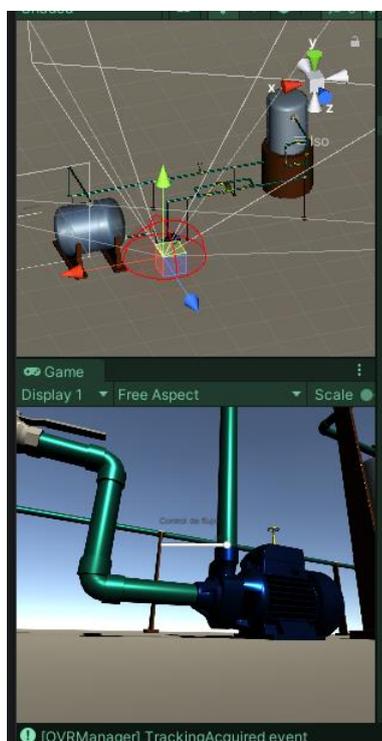


Figura 4.32 Orientación en el entorno virtual.

4.7. Visualizador con Arduino.

El módulo de Arduino estará compuesto por una pantalla LCD 16x2 la cual permitirá observar los datos de lectura del sensor de flujo y el valor seteado dentro del entorno virtual además se dispone de dos indicadores booleanos que nos indicará que el sistema está en funcionamiento.

La implementación del circuito para poder controlar el sistema del módulo de visualización con el Arduino será controlada por una comunicación serial dentro del ambiente de programación como se muestra en la figura 4.33.

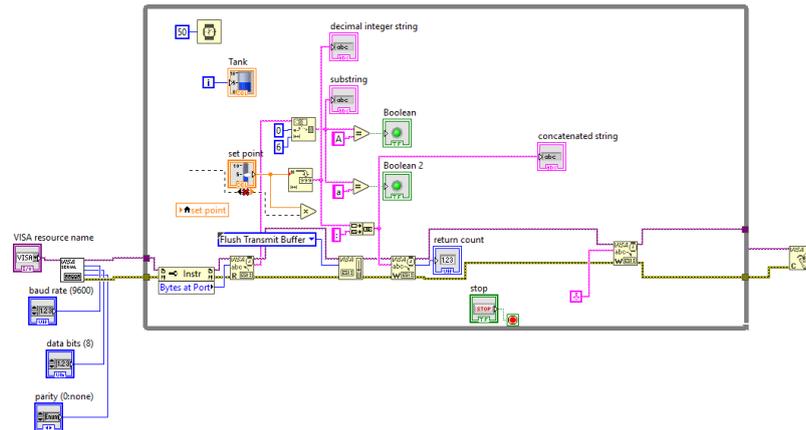


Figura 4.33 Comunicación serial del visualizador de arduino.

Donde se toma la señal proveniente del sistema de control de flujo además se toma dos indicadores led que nos indicará que el módulo de inmersión se encuentre en un estado de funcionamiento, un segundo led que indicara el estado de la bomba.

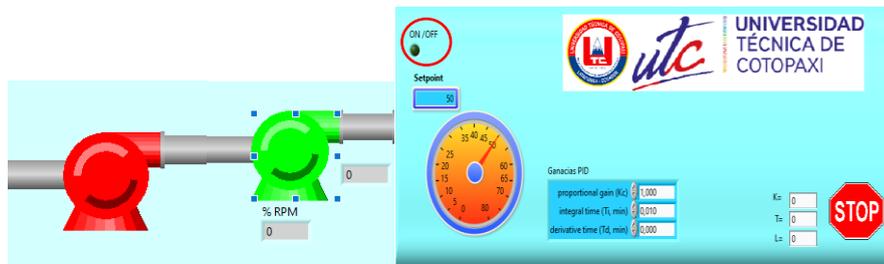


Figura 4.34. Indicadores led

5. ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS

El módulo de control de flujo se puede validar al comprobar el correcto funcionamiento cuando se manipule las diferentes variables del sistema.

5.1. Respuesta del sistema ante la variación de Set point.

La curva representada en los tiempos de establecimiento, el porcentaje de sobre impulso y valor porcentual aplicado a la bomba.

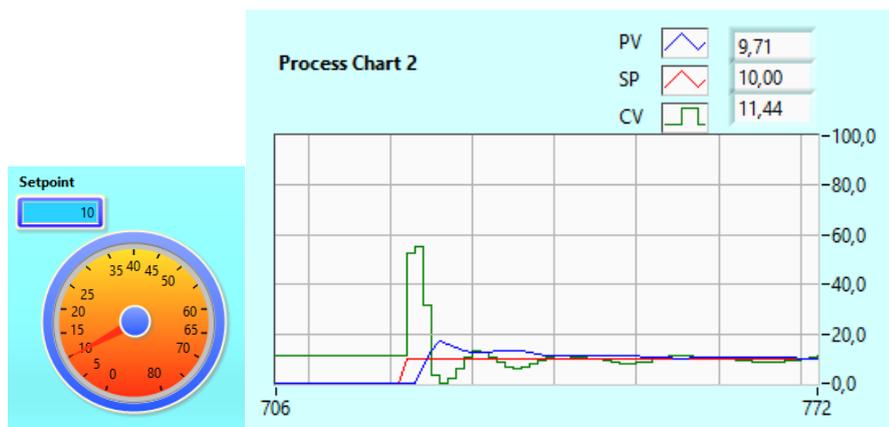


Figura 5.1 Curva de funcionamiento del sistema en 10 L/min.

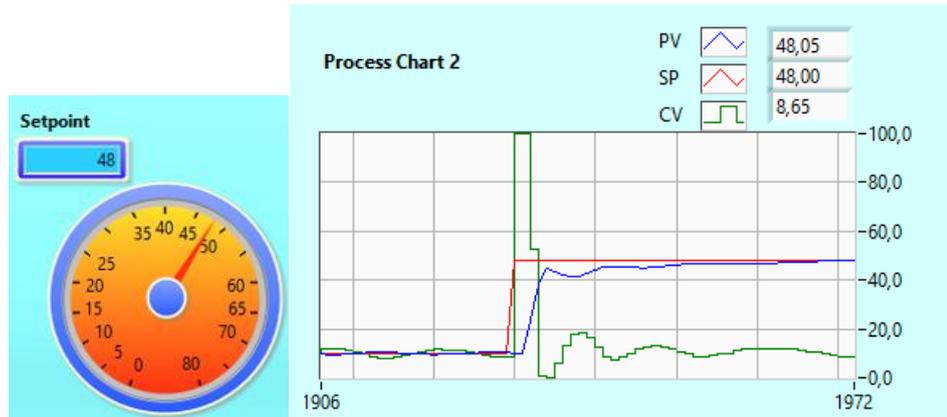


Figura 5.2 Curva de funcionamiento del sistema en 48 L/min.

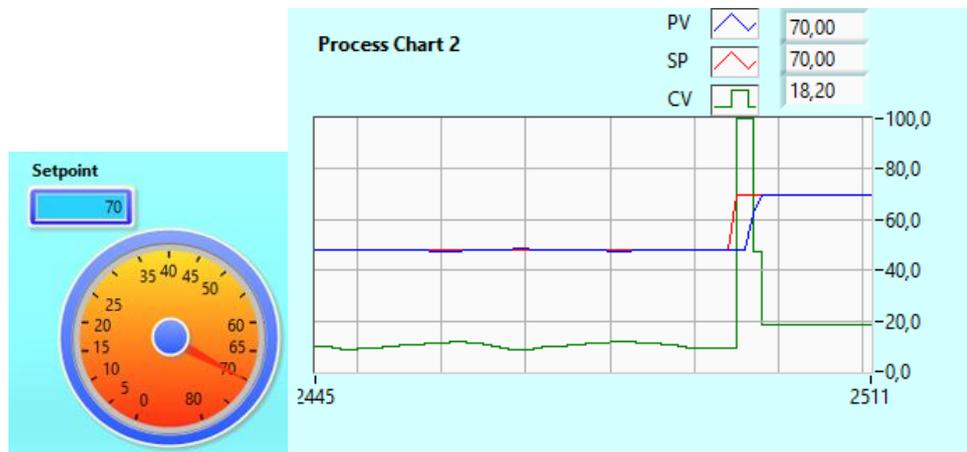


Figura 5.3 Curva de funcionamiento del sistema en 70 L/min.

Tabla 5.1 Resultados de porcentajes de sobre impulso y velocidad.

SET POINT	TIEMPO DE ESTABLECIMIENTO	% SOBRE IMPULSO	% VELOCIDAD
10	44ms	61.5%	11.4
48	39ms	0.104	15.67
70	38ms	0,25	18.2

Del análisis de los datos en la tabla se puede concluir que al momentos de iniciar el funcionamiento del módulo e ingresar el setpoint inicial, el sistema presenta una gran inestabilidad ya que se indica un gran porcentaje de sobre impulso y mayor tiempo de establecimiento debido que en la función de transferencia, el sistema presentaba un tiempo muerto que representa el tiempo que tarda al distribuirse el fluido por la tubería hacia el tanque elevado pero conforme avanza el tiempo el controlador PID reduce estos parámetros optimizando el sistema.

5.2. Respuesta del sistema ante la presencia de perturbación.

Las curvas siguientes muestra la reacción del controlador PID ante la presencia de perturbaciones por el efecto de manipular el rango de apertura de las válvulas en el sistema.

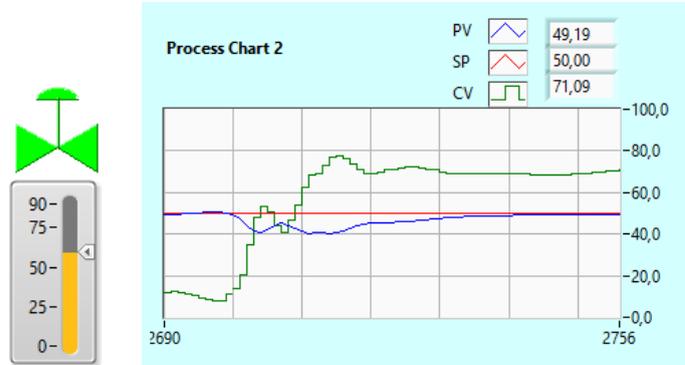


Figura 5.4 Perturbación en válvula estrangulada en 58°.

La figura 5.4 muestra que al dar una perturbación en 58 grados de apertura de la válvula que se encuentra al lado de ingreso del sensor digital indica que existe un variación de 20% del valor seteado, además presenta un tiempo de establecimiento de 44 ms para recuperar el valor de set point, y claramente se observa que el PID aumenta el porcentaje de velocidad hasta un 71.09% para poder mantener la estabilidad del sistema.

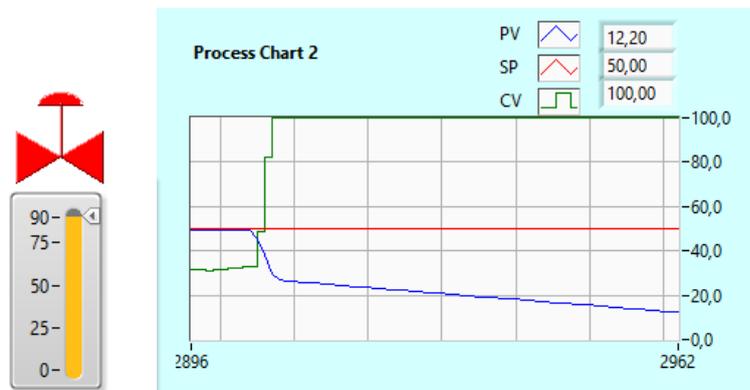


Figura 5.5 Perturbación en válvula estrangulada en 90°.

La figura 5.5 muestra que al dar una perturbación en 90 grados de apertura de la válvula que se encuentra al lado de ingreso del sensor digital indica que se pierde el control ya que el flujo no puede circular, haciendo que el dato registrado en sensor digital tiende a cero por más que el controlador mande su 100% de velocidad.

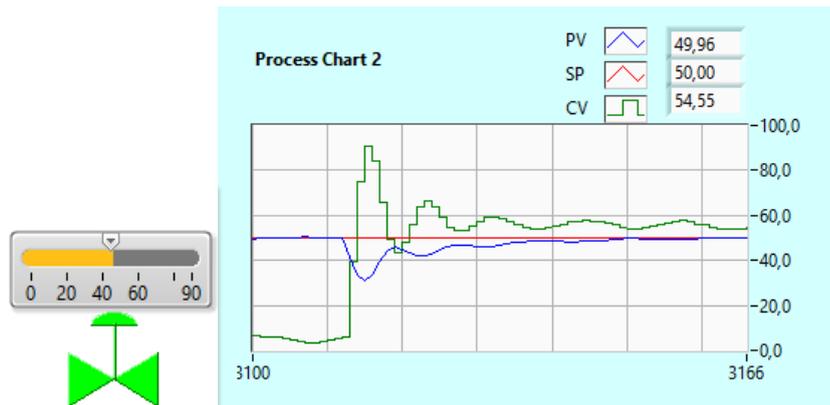


Figura 5.6 Perturbación en válvula estrangulada en 49°.

La figura 5.6 muestra que al dar una perturbación en 49 grados de apertura de la válvula de bypass en el sensor digital, indica que existen una variación de máximo 40% del valor seteado, además presenta un tiempo de establecimiento de 55 ms para recuperar el valor de set point, y claramente se observa que el PID aumenta el porcentaje de velocidad a hasta un 54.55% para poder mantener la estabilidad del sistema.

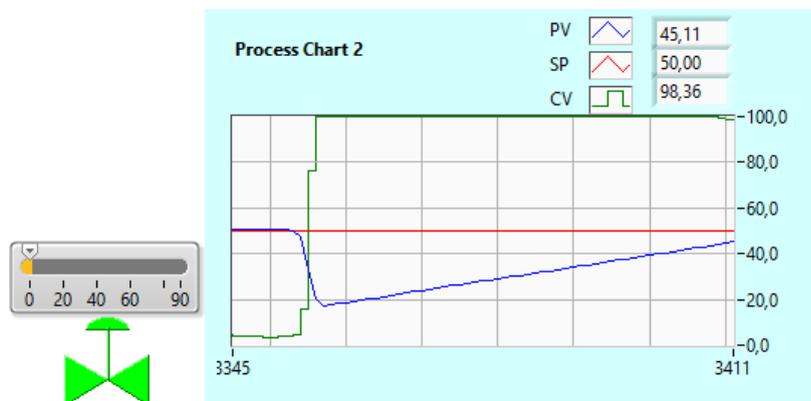


Figura 5.7 Perturbación en válvula estrangulada en 0°.

La figura 5.7 muestra que a diferencia de la anterior válvula una perturbación 100% de apertura de la válvula bypass del sensor digital indica que existen una variación de máxima del 62% del valor seteado, además presenta un tiempo de establecimiento de 71 ms para recuperar el valor de set point, y claramente se observa que el PID aumenta el porcentaje de velocidad a hasta un 100% para poder mantener la estabilidad del sistema.

5.3. Análisis del entorno virtual

Dentro del entorno virtual se emula un sistema didáctico de un control de flujo similar al módulo de banco de pruebas existente en el laboratorio de la Instrumentación y Control. La manipulación de datos se encuentra distribuido en el envío y recepción por medio de un protocolo de comunicación TCP/IP entre LabVIEW y Unity 3D.

El HMI desarrollado en Unity permite la manipulación del set point que es enviado a LabVIEW y a la vez nos devolverá la respuesta del controlador y el flujo como se observa en la figura 5.8.



Figura 5.8 HMI del set point.

Se puede observar que el nivel del tanque elevado va a variar el nivel del líquido en aumento según se el valor seteado del flujo hasta que se llene, una vez alzado el máximo nivel se abrirá la válvula de desfogue haciendo que recircule el nivel manteniendo el flujo constante.

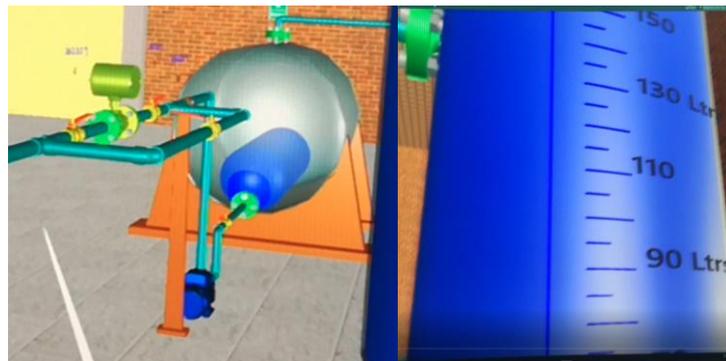


Figura 5.9. Tanques de depósito y elevado.

De igual forma el tanque de depósito se podrá observar como disminuye el nivel hasta llegar a un punto donde se mantendrá constante por la delimitación de flujo del tanque de elevación. El sistema también cuenta con la inmersión directa con los objetos del sistema como son la válvula como podemos observar en la figura 5.10.

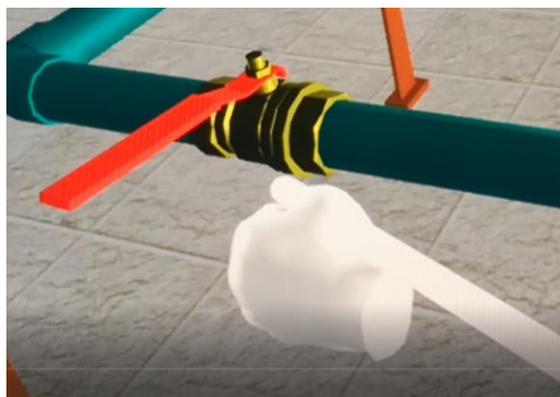


Figura 5.10. Movimiento de válvulas.

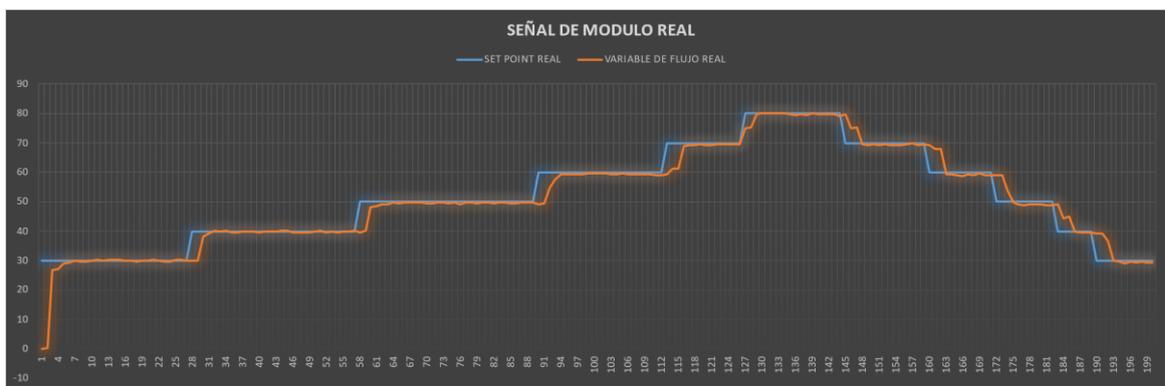
La manipulación del objeto de la válvula crea perturbaciones en el sistema haciendo que el valor seteado varíe, a continuación se presenta un primer caso donde giramos la válvula al su máximo valor que sería 90 grados esto causa una respuesta particular al sistema ya que la válvula manipulada se encuentra ubicada en el lado del ingreso de flujo al sensor lo que hace que se obstaculice el paso de fluido, haciendo que los valores en el sensor descendieran a 0 lts/min como se observa en la figura 5.11.



Figura 5.11. Valores descendiendo a 0 lts/min.

5.4. Validación de resultado respecto al Modulo Real

Para poder validar los resultados obtenidos en la simulación del control se hace una comparativa con el funcionamiento del módulo ubicado en el laboratorio de Instrumentación y Control y el módulo virtual inmersivo de control de flujo, donde se realiza un data de los valores de lectura para cada módulo variando el set point en 30, 40, 50, 60, 70,80 lts/min, donde se observan las gráficas de respuesta de la variable del proceso. Como se muestra en la siguiente figura 5.12.



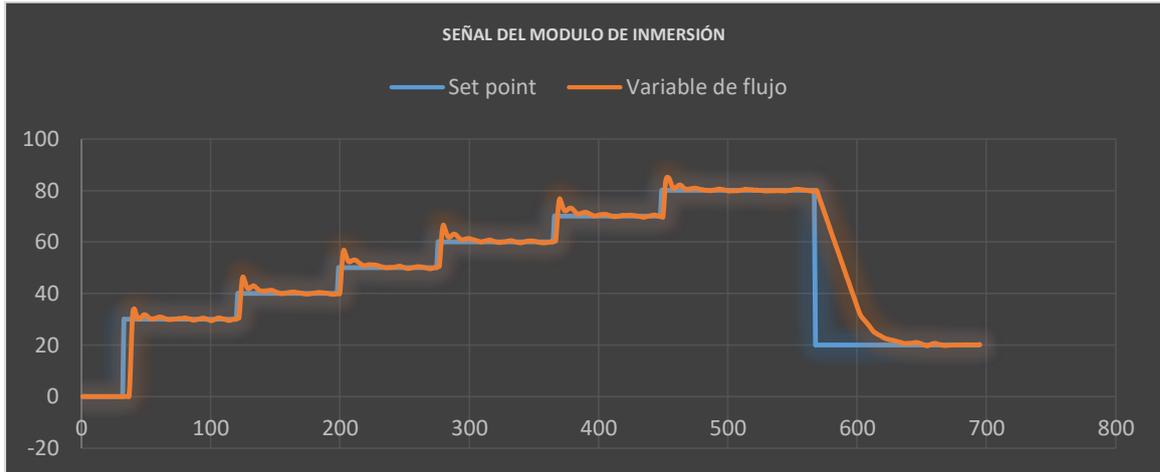


Figura 5.12. Tiempo de muestreo de 1 seg.

En las siguientes graficas se presenta un grupo de datos obtenido al momento de variar el set point en 30 lts/min, 40 lts/min, 60 lts/min, 80 lts/min donde se ve reflejado las cuatro señales que sería el set point real, set point modulo, variable flujo real y variable flujo modulo.

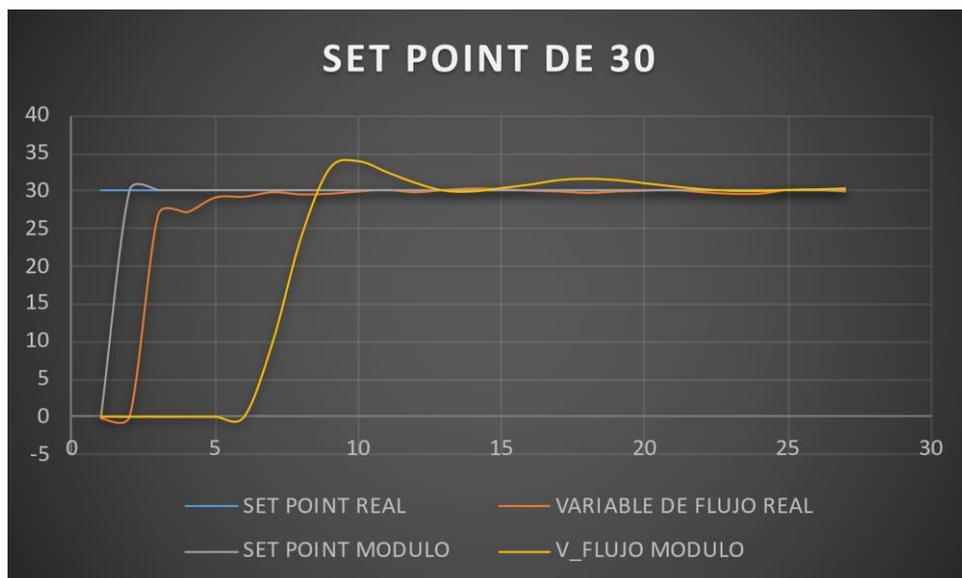


Figura 5.13. Señales del módulo real y virtual.

Al poner un set point de 30 lts /min se observa que cuando iniciamos la lectura en el primer instante partiendo del estado de reposo tanto del módulo real como del módulo virtual se determina que el módulo virtual presenta un mayor retardo para empezar a variar los valores lo cual es justificable ya que en la función de transferencia se multiplica por una función de retardo y también mediante la codificación en LabVIEW, se da un tiempo muerto a modo de simular el tiempo que se demora un fluido en llegar al punto de control.

En el caso de variar el set point a 40 lts/min ya con el sistema encendido se puede observar en la gráfica, que los tiempos de retardo se igualan además los valores en cada segundo de tiempo varía esto se debe a que estos valores van a depender del tipo de controlador que esté aplicando

en cada módulo, y a su vez cada controlador depende del valor de sintonización de sus constantes.

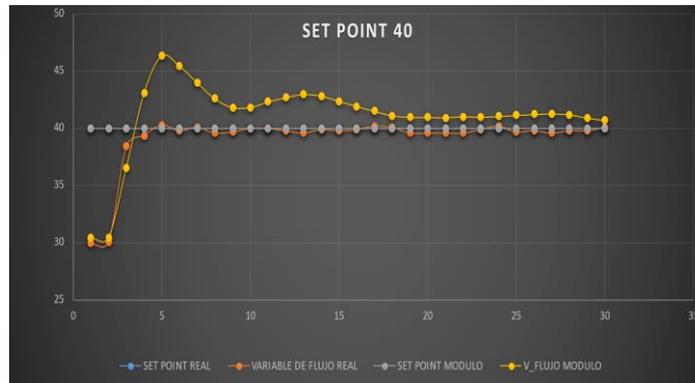


Figura 5.14. Señal del módulo real y virtual con set point 40.

Con un set point de 60 Lts/min se observa que conserva las mismas características detalladas anteriormente además se puede determinar el porcentaje de sobre impulso que el caso del módulo real no se presenta, pero en el módulo virtual si lo puede visualizar y calcular siendo este de un 25% también se puede determinar que los dos sistemas a los 18 segundos alcanza su estabilidad y aquí ya se compara que los valores son casi simulares existiendo un error de 0.01 entre el valor real y valor simulado como se muestra en la ecuación, de igual forma podemos observar la misma tendencia de resultado en el valor seteado de 80lts/min

Valores en el segundo 20

$$Error = \left| \frac{valor\ real - valor\ simulado}{valor\ real} \right| = \left| \frac{59.49 - 60.033}{50.49} \right| = 0.00912 \quad (5.1)$$

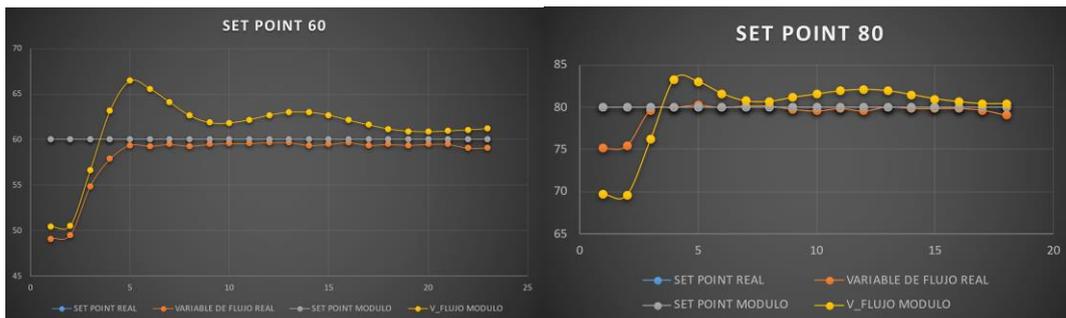


Figura 5.15. Señal del set point 60 y set point 80.

En la figura 5.16 se puede visualizar las cuatro variables en un solo diagrama observado que los valores en estado estable mantiene la tendencia de un error de 0.01 dado como resultado que el módulo virtual inmersivo de control de flujo tiene un alto grado de similitud al módulo real con el 99.99%.

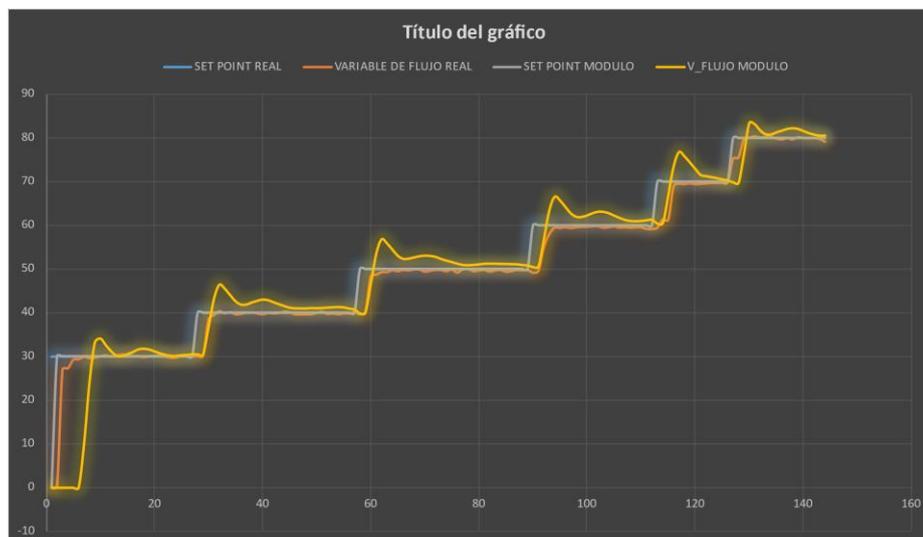


Figura 5.16. Resultado de la comparación del módulo real y módulo inmersivo.

6. PRESUPUESTO Y ANÁLISIS DE IMPACTOS

6.1. Presupuesto

Este consiste en implementar los gastos que intervinieron en la ejecución del proyecto, en las tablas que se presentan a continuación se fijan los precios de los materiales a utilizar en el proyecto.

6.1.1. Materiales y equipos electrónicos

Los materiales utilizados para el desarrollo del módulo virtual inmersivo de flujo son:

Tabla 6.1 Materiales y equipos Electrónicos.

DETALLE	DESCRIPCIÓN	CANTIDAD	VALOR UNITARIO	VALOR TOTAL
CPU	16 Gb Ram, Procesador Intel Core I5, 240 Gb de capacidad, Tarjeta gráfica Geforce GTX 1650 ultra.	1	660.71 \$	660.71 \$
Monitor Quasad	Pantalla de 19.5” Led HDMI full Hd.	1	88.57 \$	88.57 \$
Arduino Uno	Arduino Uno R3	1	12.00 \$	12.00 \$
Adaptador	Display Port a	1	8.00 \$	8.00 \$

	HDMI			
Pantalla LCD display	Pantalla Display 16x2	1	7.99 \$	7.99 \$
Gafas de Realidad Virtual	Oculus Rift	1	300.00\$	300.00\$
Módulo de semáforo led	Led rojo - verde	1	10.00\$	10.00\$
			TOTAL	1087.27 \$

6.1.2. Gastos de estructura y diseño de ingeniería del módulo virtual

La estructura del visualizador externo del módulo virtual fue diseñada por una impresora 3D.

Tabla 6.2 Estructura y diseño de Ingeniería.

DETALLE	DESCRIPCIÓN	CANTIDAD	VALOR UNITARIO	VALOR TOTAL
Estructura de caja	Dimensiones	1	20.00 \$	20.00 \$
Programación Unity	Código de programación	1	60.00 \$	60.00 \$
			TOTAL	80.00 \$

6.1.3. Gastos Varios

Son los gastos varios realizados durante el desarrollo del módulo virtual de control de flujo.

Tabla 6.3 Gastos varios.

DETALLE	DESCRIPCIÓN	VALOR TOTAL
Imprevistos	Pasajes, consultas, etc.	30.00 \$
TOTAL		30.00 \$

6.1.4. Gastos directos totales

Tabla 6.4 Gastos directos totales.

GASTOS DIRECTOS	COSTOS
Materiales y equipos electrónicos.	1087.27\$

Estructura del visualizador y diseño de ingeniería.	80.00 \$
TOTAL	1167.27\$

6.1.5. Gastos indirectos totales.

Tabla 6.5 Gastos indirectos totales.

GASTOS INDIRECTOS TOTALES	COSTOS
Imprevistos	30.00 \$
TOTAL	30.00 \$

6.1.6. Gastos totales del módulo virtual

Tabla 6.6 Gastos totales.

Gastos directos totales	1167.27\$
Gastos indirectos totales	30.00 \$
TOTAL	1197.27\$

6.2. Análisis de impacto

6.2.1. Impacto tecnológico

La tecnología sigue avanzando con el pasar del tiempo dentro del área de instrumentación y control y por ende se debe ir a la par desarrollando módulos virtuales inmersivos que sirven de mucho para la educación virtual cuando suceden situaciones de emergencia como la que está pasando actualmente el mundo entero con el Covid 19. Los estudiantes podrán observar cómo funciona todo el proceso de control de flujo mediante los sensores y la programación.

6.2.2. Impacto práctico

El módulo virtual inmersivo será de gran ayuda para los estudiantes que no puedan acceder a los laboratorios de Instrumentación por múltiples motivos y que requieran prácticas de procesos de control.

6.3. Beneficiarios del proyecto.

6.3.1. Beneficiarios directos

El presente proyecto de titulación tiene como beneficiarios directos a los estudiantes que requieran realizar prácticas de procesos de control y que no pueden acceder a los laboratorios por la actual emergencia sanitaria.

6.3.2. Beneficiarios indirectos

El proyecto tiene como beneficiarios indirectos a toda la comunidad académica al investigar nuevas técnicas de educación virtual en el área de instrumentación y control.

7. CONCLUSIONES Y RECOMENDACIONES

7.1. Conclusiones

- El módulo virtual inmersivo de control de flujo es muy similar en su funcionamiento al módulo real del laboratorio ya que los resultados de los datos muestran que a lo largo del tiempo los dos flujos se estabilizan siendo casi similares con un pequeño error del 0.01%.
- El controlador PID reacciona de manera positiva a las perturbaciones realizadas mediante las válvulas manuales manteniendo el flujo estable en todo momento.
- La simulación de control PID implementado para el entorno virtual inmersivo proporciona una aproximación a un sistema de control y concede un entrenamiento continuo y realista en ausencia de un módulo didáctico.
- La comunicación tcp/ip que conecta Unity y LabVIEW es la más adecuada porque permite un envío y recepción de datos fiables en tiempo real.
- El entorno virtual inmersivo tiene un funcionamiento estable debido a dos factores, la comunicación tcp/ip y la tarjeta de video de alta potencia gráfica, esto hace que el sistema varíe entre 110 – 135 fps.

7.2. Recomendaciones

- Si en algún momento es necesario cambiar la función de transferencia del módulo virtual inmersivo se lo puede ingresar en la programación de LabVIEW siempre y cuando esté validada.
- El proceso de control de flujo trabaja en el rango de 0 a 80 litros/minuto y si se agrega una función de transferencia que tenga otro rango de funcionamiento este va a tener problemas y no funcionará de una manera adecuada.

- Si el ordenador dispone de internet se debe dirigir a la configuración de red para obtener la dirección ip, en caso no de no tener acceso a internet se debe dirigir a la red no identificada para poder obtener la dirección ip el cual servirá para la comunicación entre Unity y LabVIEW.
- Al momento de configurar las gafas Oculus de realidad virtual se debe tener en cuenta el área de trabajo mínimo de 2x2 metros para evitar posibles accidentes.
- Para la conexión entre las gafas Oculus y el ordenador se necesita equipo de alta gama que contenga una tarjeta de video de alta potencia grafica que soporte realidad virtual.

8. REFERENCIAS

- [1] L. I. Urquiza Mendoza, B. A. Auria Burgos, S. K. Daza Suárez, F. D. R. Carriel Paredes, and R. I. Navarrete Ortega, “Uso de la realidad virtual, en la educación del futuro en centros educativos del Ecuador,” *J. Sci. Res. Rev. Cienc. e Investig.*, vol. 1, no. 4, pp. 26–30, 2016, doi: 10.26910/issn.2528-8083vol1iss4.2016pp26-30.
- [2] A. B. Sevilla, “Realidad aumentada en Educación,” p. 2, 2017, [Online]. Available: http://oa.upm.es/45985/1/Realidad_Aumentada__Educacion.pdf.
- [3] U. Politécnica, S. E. Vergara, J. ; Molina, and R. ; Machado, “Ingenius. Revista de Ciencia y Tecnología,” 2011, [Online]. Available: <http://www.redalyc.org/articulo.oa?id=505554810003>.
- [4] D. A. Armada, “Conexión de sensores usando Arduino en un entorno virtual creado con el motor Unity,” p. 33, 2019, [Online]. Available: <http://castor.det.uvigo.es:8080/xmlui/handle/123456789/405>.
- [5] A. C. Lechtaler, C. F. Blanc, M. L. Carden, A. Köhler, A. G. Polak, and J. M. Señorino, “Simulación Inmersiva con Realidad Aumentada,” 2014.
- [6] E. J. Gonzáles, “Desarrollo y caracterización de un sistema de control PID virtual para los procesos de control de nivel y flujo de un tanque de agua con la plataforma FluidLab-Pa,” Universidad de San Carlos de Guatemala, 2016.
- [7] L. M. U. Hernan, “Diseño y Construcción de un sistema didactico de Control de Caudal,” *Esc. Politécnica Nac.*, 2000, [Online]. Available: <https://bibdigital.epn.edu.ec/bitstream/15000/10687/1/T1606.pdf>.
- [8] A. Cedeño, “Módulo didáctico para controlar nivel y caudal de agua, mediante sistema SCADA, PLC y algoritmo PID.,” vol. 4, p. 13, 2019, [Online]. Available: <https://revistas.utm.edu.ec/index.php/Riemat/article/view/2196>.
- [9] José Aldamar Muñoz Hernández, *Control Automático I: Estrategias de Control*

- Clásico*. Ibagüe: Universidad de Tolima, 2014.
- [10] M. King, *Process Control: A Practical Approach*. 2010.
- [11] Guillermo Ávalos Arzate, Marcela Adriana Montúfar Navarro, Fernando Elí Ortiz Hernández, and Gustavo Villalobos Ordas, *Teoría de Control, Ajuste de Controladores Industriales*, Primera. Mexico: Instituto Politécnico Nacional, 2002.
- [12] M. A. Golato, “Características De Un Proceso,” no. Plan 2004, pp. 1–46, 2004, [Online]. Available: https://catedras.facet.unt.edu.ar/sistemasdecontrol/wp-content/uploads/sites/101/2016/03/3_Características-de-un-proceso_Análisis-de-la-respuesta-transitoria_2016.pdf.
- [13] O. Camacho, “2.30 Sliding Mode Control in Process Industry O. CAMACHO (2005),” no. January 2005, 2015.
- [14] SMAR Technology company, “Medición de flujo,” 2020, 2020. <https://www.smar.com/espanol/articulos-tecnicos/medicion-de-flujo>.
- [15] C. G. Sergio, “Modelo matemático de un tanque de nivel,” 2020. <https://controlautomaticoeducacion.com/analisis-de-sistemas/tanque-nivel/> (accessed Jan. 29, 2021).
- [16] S. Alejandro, “Que es la Realidad Virtual,” 2019, 1990. www.artfutura.org/v2/pasteditions.php?idcontent=8&mb=2.
- [17] M. Ojanguren Álvarez and Marina, “Realidad virtual en la ingeniería civil : virtualización de una obra,” 2016, [Online]. Available: <https://repositorio.unican.es/xmlui/handle/10902/10018>.
- [18] D. Levis, “¿Qué es la realidad virtual?,” *Diegoelvis*, pp. 1–28, 2006, [Online]. Available: <http://www.diegolevis.com.ar>.
- [19] Ecured, “Autodesk Inventor,” 2010. https://www.ecured.cu/Autodesk_Inventor (accessed Jan. 03, 2021).
- [20] U. Technologies, “Unity 3D,” 2020, [Online]. Available: <https://unity.com/es>.
- [21] J. César and O. López, “Visión estereoscópica en sistemas de visualización inmersiva – Ejemplos prácticos,” vol. 8, pp. 96–106, 2011.
- [22] P. Sarhadi and S. Yousefpour, “State of the art: hardware in the loop modeling and simulation with its applications in design, development and implementation of system and control software,” *Int. J. Dyn. Control*, vol. 3, no. 4, pp. 470–479, 2015, doi: 10.1007/s40435-014-0108-3.
- [23] IMC, “Hardware in the loop,” 2020. <https://www.imc-tm.mx/aplicaciones/soluciones->

generales/simulacion-hil/.

- [24] Rubén Morales Menéndez y Ricardo Ambrosio Ramírez Mendoza, “Sistemas de control moderno volumen I: sistemas de tiempo continuo,” vol. 1, no. 9, pp. 1–29, 2013, [Online]. Available: <http://prod77ms.itesm.mx/podcast/EDTM/ID295.pdf>.
- [25] I. Rueda F, “Implementación de un controlador embebido para controlar una mesa XY,” *Pontif. Univ. Católica Del Ecuador*, vol. 96, pp. 179–200, 2013, [Online]. Available:
http://tesis.pucp.edu.pe/repositorio/bitstream/handle/20.500.12404/6097/ACOSTA_CARLOS_DISEÑO_MAQUINA_REBANADORA.pdf?sequence=1.
- [26] G. H. M. De Arruda, “Controladores PID Industriais,” pp. 1–24, 2000.
- [27] S. Kouro and R. Musalem, “Control mediante lógica difusa,” *Técnicas Mod. Autom.*, pp. 1–7, 2002, [Online]. Available:
<http://www2.elo.utfsm.cl/~elo377/documentos/Fuzzy.pdf>.
- [28] E. Fernández Camacho and C. Bordóns Alba, “Control Predictivo: Pasado, presente y futuro,” *Rev. Iberoam. Automática e Informática Ind.*, vol. 1, no. 3, pp. 5–28, 2004, doi: 10.4995/riai.v1i3.10587.
- [29] N. Instruments, “LabVIEW,” *National Instruments*, 2011.
<https://forums.ni.com/t5/LabVIEW/Basic-Instrumentation-Simulation/td-p/1781952?profile.language=es> (accessed Feb. 16, 2021).

ANEXOS

Anexo 1. Programación en Unity

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class bloquoe : MonoBehaviour
{
    // Start is called before the first frame update

    public GameObject valvlula_IN_SD;
    float x;
    float y;
    float z;
    float rot_x;
    float rot_y;
    float rot_z;
    float t;

    public void posimodel()
    {
        x = Mathf.Clamp(-2.9529f, -2.9529f, -2.9529f);
        y = Mathf.Clamp(1.32665f, 1.32665f, 1.32665f);
        z = Mathf.Clamp(-2.97455f, -2.97455f, -2.97455f);
        rot_x = Mathf.Clamp(0f,0f,0f);
        rot_y = Mathf.Clamp(rot_y,0f,90f);
        rot_z = Mathf.Clamp(0,0,0);

        valvlula_IN_SD.transform.localPosition = new Vector3(x, y, z);
        valvlula_IN_SD.transform.rotation = Quaternion.Euler(rot_x, rot_y, rot_z);
    }
}

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class digital: MonoBehaviour
{
    // Start is called before the first frame update

    Text S_digitalText;

    float sd;
    void Start()
    {
        S_digitalText = GetComponent<Text>();
    }

    // Update is called once per frame

    public void textUpdate(float value)
    {
        sd = value;
        S_digitalText.text = Mathf.RoundToInt(sd)+"";
    }
}
```

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class flujo : MonoBehaviour
{
    public GameObject rotametro;
    public Slider mover;
    public GameObject AGUA;
    public Slider escalar;
    public GameObject AGUAre;
    public Slider escalarre;
    float x;
    float y;
    float z;
    float x1;
    float y1;
    float z1;
    float t;

    public void posimodel()
    {
        t = ((mover.value*(0.0031f))-0.1669f) ;
        rotametro.transform.localPosition = new Vector3(0, 0, t);
    }

    public void escalarmodel()
    {
        x = 1;
        z = 1;

        y = (mover.value ) / 70;

        AGUA.transform.localScale= new Vector3(x, y, z);
    }

    public void escalarmodelreserva()
    {
        x1 = 1;

        z1 = (mover.value*(-0.01642f)) +1.15f;

        y1 = (mover.value * (-0.01642f)) + 1.15f;

        AGUAre.transform.localScale = new Vector3(x1, y1, z1);
    }
}

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class NewBehaviourScript : MonoBehaviour
{
    // Start is called before the first frame update
    void Start()

```

```

    {
    }

    // Update is called once per frame
    void Update()
    {
    }
}

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class numero: MonoBehaviour
{
    // Start is called before the first frame update
    Text porcentajeText;

    float t;
    void Start()
    {
        porcentajeText = GetComponent<Text> ();
    }

    // Update is called once per frame
    public void textUpdate (float value)
    {
        t = value ;
        porcentajeText.text = Mathf.RoundToInt(t) + "Lts/Min";
    }
}

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.XR;

public class programa_manos : MonoBehaviour

{
    public bool showController = false;
    public InputDeviceCharacteristics controllerCharacteristicas;
    public List<GameObject> controllerPrefabs;
    public GameObject modelo_ManosPrefabs;

    private InputDevice targetDevice;
    private GameObject spawnedController;
    private GameObject spawnedmodelo_Manos;
    private Animator handAnimator;

    // Start is called before the first frame update

    void Start()
    {
        TriInitialize();

```

```

    }

    void TriInitialize()
    {
        List<InputDevice> devices = new List<InputDevice>();

        InputDevices.GetDevicesWithCharacteristics(controllerCharacteristics,
devices);

        foreach (var item in devices)
        {
            Debug.Log(item.name + item.characteristics);

        }
        if (devices.Count > 0)
        {
            targetDevice = devices[0];
            GameObject prefab = controllerPrefabs.Find(controller => controller.name
== targetDevice.name);
            if (prefab)
            {
                spawnedController = Instantiate(prefab, transform);
            }
            else
            {
                Debug.LogError("NO SE ENCUETRA EL MODELO");
                spawnedController = Instantiate(controllerPrefabs[0], transform);
            }
            spawnedmodelo_Manos = Instantiate(modelo_ManosPrefabs, transform);
            handAnimator = spawnedmodelo_Manos.GetComponent<Animator>();
        }
    }

    void UpdateHandAnimation()
    {
        if (targetDevice.TryGetFeatureValue(CommonUsages.trigger, out float
triggerValue))
        {
            handAnimator.SetFloat("Trigger", triggerValue);
        }
        else
        {
            handAnimator.SetFloat("Trigger", 0);
        }

        if (targetDevice.TryGetFeatureValue(CommonUsages.grip, out float gripValue))
        {
            handAnimator.SetFloat("Grip", gripValue);
        }
        else
        {
            handAnimator.SetFloat("Grip", 0);
        }
    }

    // Update is called once per frame
    void Update()
    {

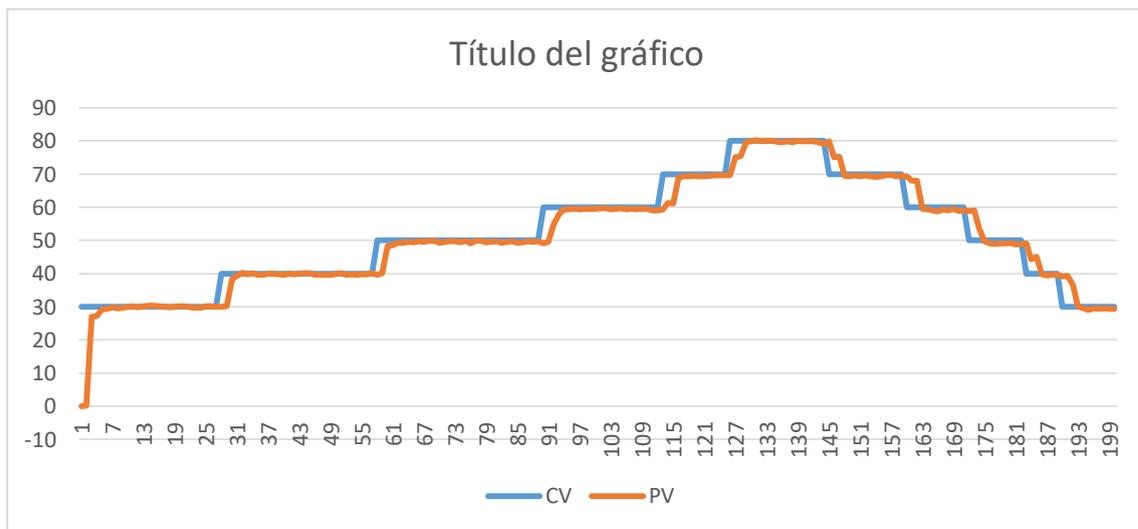
```

```
if (!targetDevice.isValid)
{
    TriInitialize();
}
else
{
    if (showController)
    {
        spawnedmodelo_Manos.SetActive(false);
        spawnedController.SetActive(true);

    }
    else
    {
        spawnedmodelo_Manos.SetActive(true);
        spawnedController.SetActive(false);
        UpdateHandAnimation();
    }
}
}
}
```




Anexo 3. Base de datos de la función de transferencia



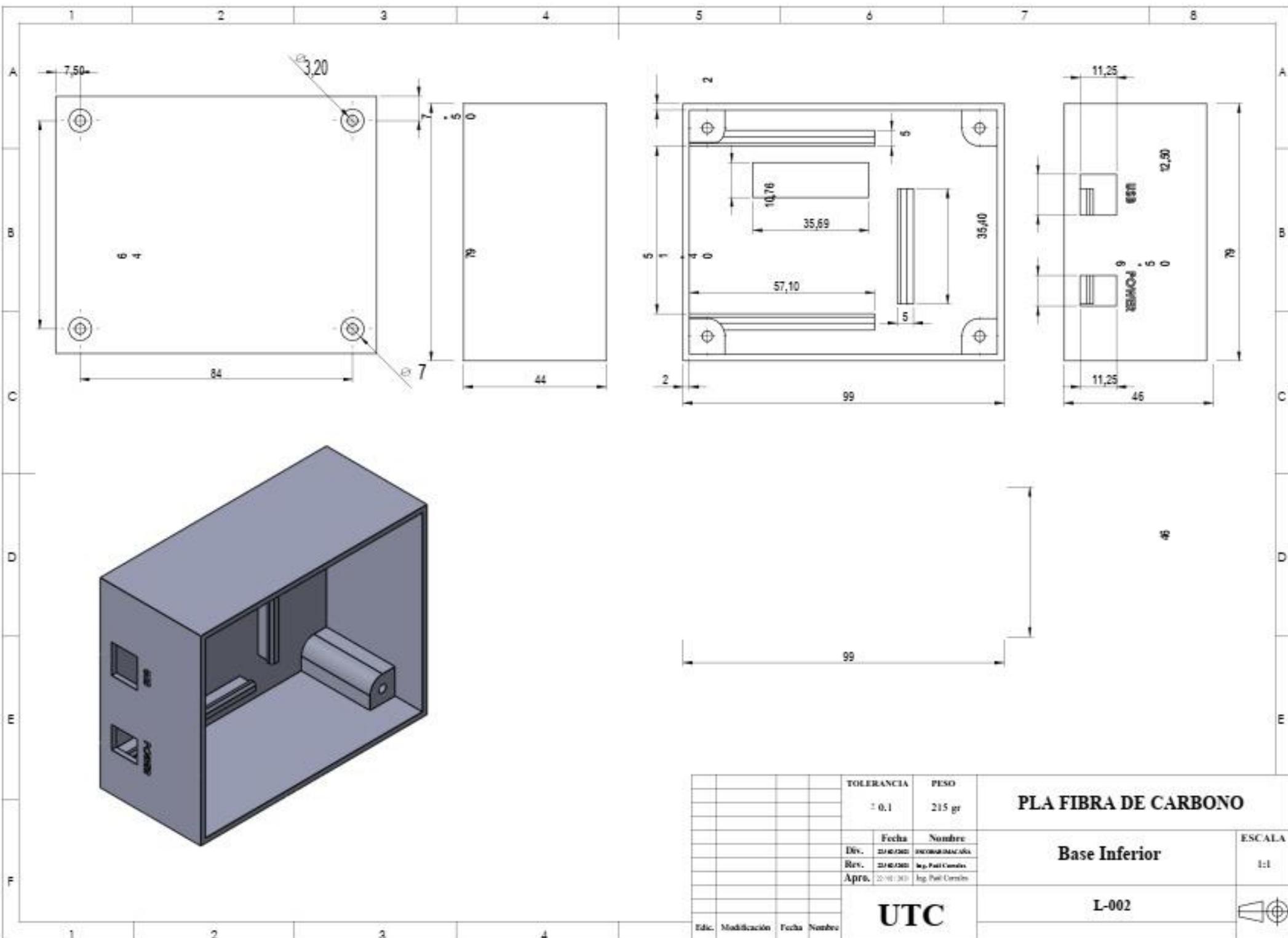
Fecha-Hora	CV	PV
04/02/21-11:33:07	30	-0,08
04/02/21-11:33:08	30	0,22
04/02/21-11:33:09	30	26,92
04/02/21-11:33:10	30	27,22
04/02/21-11:33:11	30	29,19
04/02/21-11:33:12	30	29,29
04/02/21-11:33:13	30	29,88
04/02/21-11:33:14	30	29,59
04/02/21-11:33:15	30	29,68
04/02/21-11:33:17	30	29,98
04/02/21-11:33:18	30	30,18
04/02/21-11:33:19	30	29,88

04/02/21-11:33:20	30	30,18
04/02/21-11:33:21	30	30,38
04/02/21-11:33:22	30	30,28
04/02/21-11:33:23	30	30,08
04/02/21-11:33:24	30	29,98
04/02/21-11:33:25	30	29,78
04/02/21-11:33:26	30	29,98
04/02/21-11:33:27	30	30,08
04/02/21-11:33:28	30	30,18
04/02/21-11:33:29	30	29,88
04/02/21-11:33:30	30	29,68
04/02/21-11:33:31	30	29,68
04/02/21-11:33:32	30	30,18
04/02/21-11:33:33	30	30,18
04/02/21-11:33:34	30	29,98
04/02/21-11:33:35	40	29,98
04/02/21-11:33:37	40	30,08
04/02/21-11:33:38	40	38,39
04/02/21-11:33:39	40	39,38
04/02/21-11:33:40	40	40,28
04/02/21-11:33:41	40	39,78
04/02/21-11:33:42	40	40,08
04/02/21-11:33:43	40	39,58
04/02/21-11:33:44	40	39,68
04/02/21-11:33:45	40	39,98
04/02/21-11:33:46	40	39,98
04/02/21-11:33:47	40	39,78
04/02/21-11:33:48	40	39,58
04/02/21-11:33:49	40	39,88
04/02/21-11:33:50	40	39,78
04/02/21-11:33:51	40	39,88
04/02/21-11:33:52	40	40,18
04/02/21-11:33:53	40	40,08
04/02/21-11:33:55	40	39,58
04/02/21-11:33:56	40	39,58
04/02/21-11:33:57	40	39,58
04/02/21-11:33:58	40	39,58
04/02/21-11:33:59	40	39,88
04/02/21-11:34:00	40	40,18
04/02/21-11:34:01	40	39,68
04/02/21-11:34:02	40	39,78
04/02/21-11:34:03	40	39,58
04/02/21-11:34:04	40	39,78
04/02/21-11:34:05	40	39,78
04/02/21-11:34:06	40	40,08
04/02/21-11:34:07	50	39,58
04/02/21-11:34:08	50	40,18
04/02/21-11:34:09	50	48,22
04/02/21-11:34:11	50	48,62

04/02/21-11:34:12	50	49,22
04/02/21-11:34:13	50	49,22
04/02/21-11:34:14	50	49,62
04/02/21-11:34:15	50	49,42
04/02/21-11:34:16	50	49,72
04/02/21-11:34:17	50	49,62
04/02/21-11:34:18	50	49,82
04/02/21-11:34:19	50	49,82
04/02/21-11:34:20	50	49,32
04/02/21-11:34:21	50	49,52
04/02/21-11:34:22	50	49,72
04/02/21-11:34:23	50	49,72
04/02/21-11:34:24	50	49,42
04/02/21-11:34:25	50	49,82
04/02/21-11:34:26	50	49,12
04/02/21-11:34:27	50	49,82
04/02/21-11:34:28	50	49,91
04/02/21-11:34:30	50	49,42
04/02/21-11:34:31	50	49,62
04/02/21-11:34:32	50	49,72
04/02/21-11:34:33	50	49,32
04/02/21-11:34:34	50	49,62
04/02/21-11:34:35	50	49,72
04/02/21-11:34:36	50	49,32
04/02/21-11:34:37	50	49,42
04/02/21-11:34:38	50	49,72
04/02/21-11:34:39	50	49,62
04/02/21-11:34:40	50	49,72
04/02/21-11:34:41	60	49,12
04/02/21-11:34:43	60	49,52
04/02/21-11:34:44	60	54,9
04/02/21-11:34:45	60	57,89
04/02/21-11:34:46	60	59,39
04/02/21-11:34:47	60	59,29
04/02/21-11:34:48	60	59,49
04/02/21-11:34:49	60	59,29
04/02/21-11:34:50	60	59,49
04/02/21-11:34:51	60	59,59
04/02/21-11:34:52	60	59,59
04/02/21-11:34:53	60	59,69
04/02/21-11:34:54	60	59,69
04/02/21-11:34:55	60	59,39
04/02/21-11:34:56	60	59,49
04/02/21-11:34:57	60	59,69
04/02/21-11:34:58	60	59,39
04/02/21-11:35:00	60	59,49
04/02/21-11:35:01	60	59,39
04/02/21-11:35:02	60	59,49
04/02/21-11:35:03	60	59,49

04/02/21-11:35:04	60	59,09
04/02/21-11:35:05	60	59,09
04/02/21-11:35:06	70	59,39
04/02/21-11:35:07	70	61,29
04/02/21-11:35:08	70	61,19
04/02/21-11:35:09	70	68,9
04/02/21-11:35:10	70	69,4
04/02/21-11:35:11	70	69,3
04/02/21-11:35:12	70	69,5
04/02/21-11:35:13	70	69,3
04/02/21-11:35:15	70	69,4
04/02/21-11:35:16	70	69,5
04/02/21-11:35:17	70	69,6
04/02/21-11:35:18	70	69,6
04/02/21-11:35:19	70	69,6
04/02/21-11:35:20	80	69,7
04/02/21-11:35:21	80	75,12
04/02/21-11:35:22	80	75,42
04/02/21-11:35:23	80	79,65
04/02/21-11:35:24	80	79,95
04/02/21-11:35:25	80	80,25
04/02/21-11:35:26	80	79,95
04/02/21-11:35:27	80	80,15
04/02/21-11:35:28	80	80,15
04/02/21-11:35:29	80	79,75
04/02/21-11:35:30	80	79,55
04/02/21-11:35:31	80	79,85
04/02/21-11:35:33	80	79,55
04/02/21-11:35:34	80	80,05
04/02/21-11:35:35	80	79,85
04/02/21-11:35:36	80	79,85
04/02/21-11:35:37	80	79,85
04/02/21-11:35:38	80	79,65
04/02/21-11:35:39	80	79,04
04/02/21-11:35:40	70	79,85
04/02/21-11:35:41	70	75,12
04/02/21-11:35:42	70	75,22
04/02/21-11:35:43	70	69,5
04/02/21-11:35:44	70	69,4
04/02/21-11:35:45	70	69,6
04/02/21-11:35:46	70	69,4
04/02/21-11:35:47	70	69,6
04/02/21-11:35:48	70	69,4
04/02/21-11:35:50	70	69,2
04/02/21-11:35:51	70	69,4
04/02/21-11:35:52	70	69,7
04/02/21-11:35:53	70	69,8
04/02/21-11:35:54	70	69,4
04/02/21-11:35:55	70	69,5

04/02/21-11:35:56	60	69,4
04/02/21-11:35:57	60	68,1
04/02/21-11:35:58	60	68
04/02/21-11:35:59	60	59,49
04/02/21-11:36:00	60	59,29
04/02/21-11:36:01	60	58,99
04/02/21-11:36:02	60	58,79
04/02/21-11:36:03	60	59,39
04/02/21-11:36:04	60	59,09
04/02/21-11:36:05	60	59,59
04/02/21-11:36:06	60	58,89
04/02/21-11:36:08	60	59,09
04/02/21-11:36:09	50	58,89
04/02/21-11:36:10	50	59,09
04/02/21-11:36:11	50	53,6
04/02/21-11:36:12	50	49,72
04/02/21-11:36:13	50	49,12
04/02/21-11:36:14	50	48,92
04/02/21-11:36:15	50	49,02
04/02/21-11:36:16	50	49,02
04/02/21-11:36:17	50	49,22
04/02/21-11:36:18	50	48,82
04/02/21-11:36:19	50	48,92
04/02/21-11:36:20	40	49,12
04/02/21-11:36:21	40	44,35
04/02/21-11:36:22	40	45,04
04/02/21-11:36:23	40	39,78
04/02/21-11:36:24	40	39,48
04/02/21-11:36:25	40	39,58
04/02/21-11:36:26	40	39,68
04/02/21-11:36:27	30	39,18
04/02/21-11:36:29	30	39,28
04/02/21-11:36:30	30	36,61
04/02/21-11:36:31	30	30,08
04/02/21-11:36:32	30	29,59
04/02/21-11:36:33	30	29,09
04/02/21-11:36:34	30	29,59
04/02/21-11:36:35	30	29,39
04/02/21-11:36:36	30	29,59
04/02/21-11:36:37	30	29,39
04/02/21-11:36:38	30	29,29



		TOLERANCIA	PESO	PLA FIBRA DE CARBONO	
		± 0.1	215 gr		
		Fecha	Nombre	Base Inferior	ESCALA 1:1
Div.	23/05/2020	INGENIERO MECANICO			
Rev.	23/05/2020	Ing. Paul Cornejo			
Apr.	22/05/2020	Ing. Paul Cornejo			
		UTC		L-002	
Edic.	Modificación	Fecha	Nombre		

MANUAL DE USUARIO

El presente manual corresponde al diseño y desarrollo de la plataforma virtual que pretende virtualizar un módulo didáctico de un proceso de flujo, desarrollado mediante dos etapas, la primera etapa consistió en el modelamiento de la planta en un software CAD posteriormente importada a Unity para crear un ambiente inmersivo que funcione con gafas de realidad virtual en donde la persona se involucra en el proceso a fin de analizar el comportamiento del mismo. La segunda etapa es el diseño de controladores para aplicar el control virtual de flujo. Para ello se emplea el modelo matemático enlazando el software de instrumentación con el motor gráfico de Unity para el correcto funcionamiento, adicional se implementará un visualizador externo mediante Arduino Uno para observar datos del proceso de flujo.

El flujo se comporta de manera estable en todos los valores de referencia o SetPoint que se añadió al sistema, las gafas de realidad virtual se comunican de manera efectiva con la computadora para la correcta inmersión. La comunicación serial entre los dos softwares funciona correctamente y todo el sistema tiene una eficiencia del 90 %.

Requisitos

El desarrollo de la plataforma virtual inmersiva para el entrenamiento del control de procesos de flujo, necesita de ciertas especificaciones para cumplir con el desarrollo óptimo de la aplicación y manipulación de la plataforma virtual y se debe cumplir con los siguientes requisitos tanto de hardware, software, dimensiones de trabajo.

Requisitos Hardware

Puesto que la realidad virtual usa recursos de alta gama en tecnología se definirán recursos mínimos a fin de alcanzar el correcto funcionamiento de la plataforma. Estos requisitos corresponderán a: memoria Ram, disco duro, procesador, tarjeta gráfica, etc.

Especificaciones mínimas	
Disco duro	Disco HDD de 512 Gb o Disco SSD de 512 Gb.
Memoria RAM	> 8 Gb de Ram
Tarjeta gráfica	NVIDIA GTX 1050Ti/AMD Radeon RX 470 o superior
Tarjeta gráfica alternativa	NVIDIA GTX 960 de 4 Gb/AMD Radeon R9 290
CPU	Intel i5-4590 / AMD Ryzen i5 1500X o superior
Salida de video	Salida de video HDMI 1.3
Puertos USB	Puertos USB 2.0
Sistema operativo	Windows 10

Cabe recalcar que los requisitos mínimos corresponden a un computador de gama media o un computador gamer de gama baja. Dentro de las especificaciones de hardware también se incluyen los requisitos mínimos de periféricos de entrada y salida, incluyendo los periféricos para la realidad virtual.

PERIFERICOS:

6 puertos USB

1 puerto de entrada HDMI

1 puerto Virtual Serial Port

1 teclado

1 mouse

1 pantalla

1 gafas de realidad virtual

2 mandos de orientación

2 sensores de espacio

1 módulo Arduino

Requisitos Software

Los requisitos en software de programación y desarrollo para la aplicación de la plataforma en realidad virtual creada serán específicos y corresponderán a las versiones con las que se crearon dicha plataforma.

NOTA: para su correcto funcionamiento tomar muy en cuenta las versiones de software en la que se desarrolló la plataforma y consultar si las nuevas versiones de los softwares no afectan o dañen el funcionamiento de la misma.

SOFTWARES:

Oculus Rift s

Arduino IDE 1.8.13

LabView 2020 SP1 32 bits

Unity 2019.4.14f1 64 bits

Visual Studio 2019

NOTA: la conexión a internet para la plataforma no es necesaria puesto que la misma no usa internet.

Requisitos de Trabajo

La zona de trabajo dentro la plataforma desarrollada es muy esencial debido a que si la zona en

cuestión no está despejada al momento de usar las gafas de realidad virtual la persona que está operando la plataforma podría sufrir graves lesiones.

Las dimensiones recomendadas para medir la zona de trabajo son 2 metros, este espacio será el mínimo necesario para remarcar la zona de trabajo de las gafas en realidad virtual.

La separación mínima entre los sensores de espacio – movimiento deberán tener una distancia como mínimo de 1.5 mts, y los mismos deberán estar de una forma horizontal uno referente al otro (los sensores de espacio - movimiento por lo general denotan la ubicación y distancia que toma el casco de realidad virtual con respecto al CPU donde se conectan).

Los mandos de las gafas de realidad virtual denotaran en si el trabajo con respecto al casco y los mismos se manejara con las manos del participante.

NOTA: para una mejor experiencia en el entorno de realidad virtual se recomienda manejar los mandos de la realidad virtual con las manos del usuario, ayudando a movilizarse y manipular el entorno de la plataforma diseñada.



Ilustración 1: Medidas y zona de trabajo para la manipulación de la realidad virtual.

Programas de desarrollo

Los programas que se van a usar para la manipulación del entorno de realidad virtual se describen en la sección 1.2 del manual. En esta sección se describirán a breves rasgos las

características de los programas a usarse.

Arduino IDE

El software para la programación de Arduino en cualquiera de sus versiones (uno, mega, nano, etc.), se lo conoce como Arduino IDE esta plataforma es de dominio libre, su descarga se la puede realizar desde su página oficial <https://www.arduino.cc/en/software>.

NOTA: dentro de la página oficial de Arduino se puede seleccionar la versión con la que se requiere trabajar, para más detalles de la versión consultar la sección anterior, requisitos de software.

Para la instalación del software, una vez descargado ubicamos la carpeta de descarga y damos doble clic sobre el icono de aplicación, concedemos todos los permisos de usuario y aceptamos a todos los pasos de instalación.

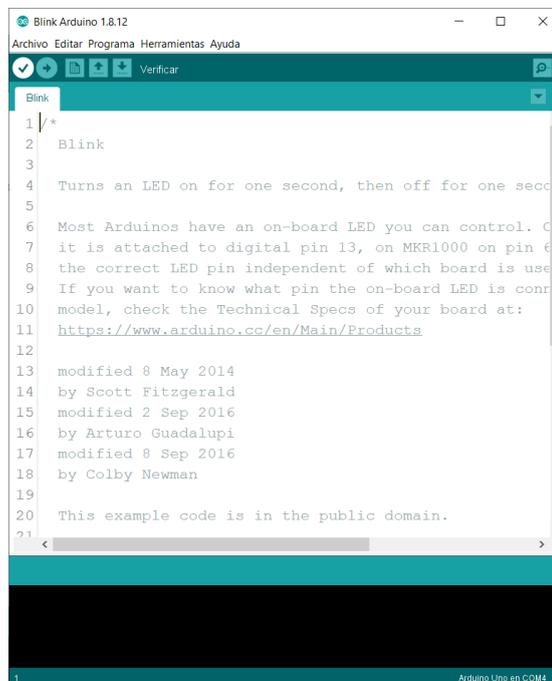


Ilustración 2: Programa Arduino IDE

LabView

Para la instalación de LabView ingresaremos a la página oficial del software llamada NATIONAL INSTRUMENTS <https://www.ni.com/es-cr/support/downloads/software-products/download.labview.html#369643>. Aquí descargaremos el software para el desarrollo de la interfaz gráfica y visualización.

NOTA: tomar en cuenta que el programa es de software propietario y que la persona que va a manipular el mismo debe poseer una cuenta activa dentro del software, si la universidad es la que va a manejar la plataforma, consultar si la misma dispone de licencia activa para el uso libre del software. Los desarrolladores del presente proyecto no se responsabilizan si los

usuarios que manipulan la plataforma se descargan softwares piratas.

El software es de fácil instalación, daremos permisos de administrador al software y seleccionaremos las opciones de aceptar y siguiente hasta terminar con la instalación del mismo.

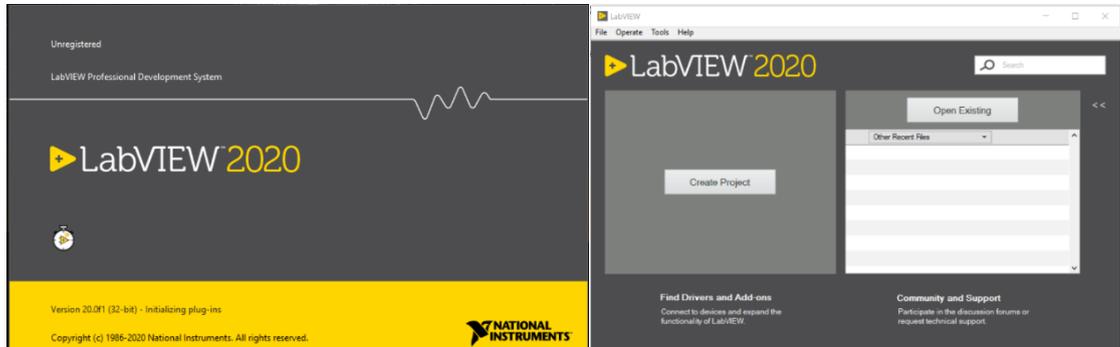


Ilustración 3: interfaz gráfica LabView 2020

Unity 2019

Para la instalación del software Unity ingresaremos a la página oficial de Unity <https://store.unity.com/es/#plans-individual>, para la posterior descarga del software a usar, dentro de la plataforma se puede seleccionar la versión a descargar (para ver la versión que se puede usar consultar la sección anterior, requisitos software), descargamos la versión a usar y seguimos los pasos de instalación.

NOTA: el software Unity es un software libre de restricciones al uso. Cuando la persona que usa este software lanza en línea o vende su aplicación esta no debe de exceder a una ganancia de cien mil dólares anuales, si el usuario considera que su programa va a exceder el valor libre de ganancia otorgado por los desarrolladores de la aplicación deberá pagar licencias y aplicaciones. Para más información leer la página oficial de Unity.

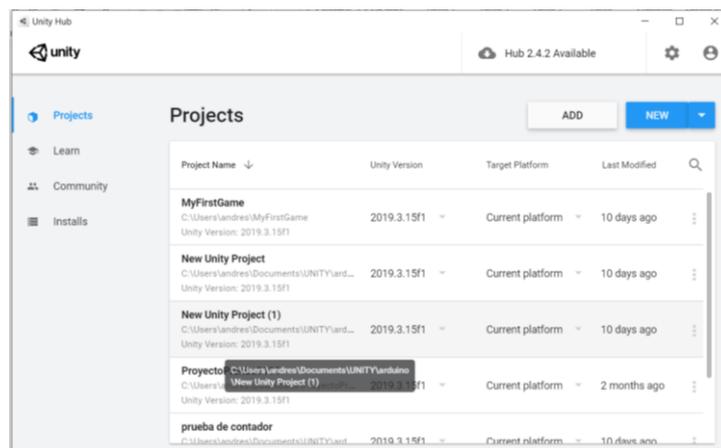


Ilustración 4: Interfaz gráfica Unity

Visual Studio

El programa con el que se definirá y dará programación en la interfaz de Unity será Visual Studio, este software es una plataforma de programación orientada a objetos con programación

fuente en C++. Su descarga se la puede realizar desde su página oficial y es de carácter libre <https://visualstudio.microsoft.com/es/downloads/>.

Para la instalación de la plataforma de programación se le dará permisos de usuario y luego aceptaremos y daremos en siguiente a todas las indicaciones establecidas por dicho software.

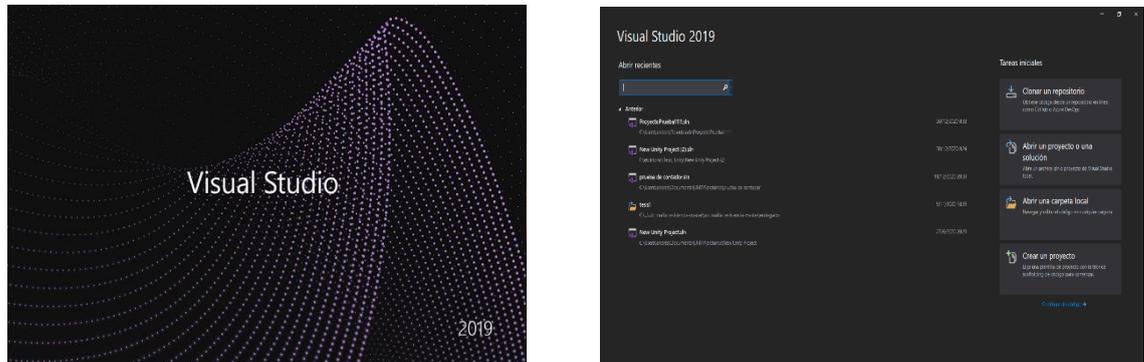


Ilustración 5: Entorno de programación en Visual Studio

Oculus Rift

Oculus es un software propietario y definido por las gafas de realidad virtual que se van a emplear. Para la instalación del software Oculus ingresaremos a la página oficial de las gafas virtuales <https://www.oculus.com/setup/#rift-s-setup>, dentro de la plataforma se puede seleccionar la versión a descargar (para ver la versión que se puede usar consultar la sección anterior, requisitos software), descargamos la versión a usar y seguimos los pasos de instalación.



Ilustración 6: Entorno gafas de realidad virtual Oculus

Configuración Realidad Virtual

Configurar Rift

Para la configuración de las gafas Oculus Rift se tendrá que tener todos los periféricos conectados al CPU, para esto se seguirá los siguientes pasos:

Prepárate para conectar el visor de Rift y el sensor de Oculus a tu ordenador.

Conecta el extremo HDMI del cable del visor al puerto HDMI de tu tarjeta gráfica.

NOTA: No utilizar el puerto HDMI de la placa base, si lo hubiera. Si no se tiene claro qué puerto HDMI se debe utilizar, probar con el que está en el panel más estrecho y más simple en la parte trasera del ordenador. (La mayoría de las tarjetas gráficas miden alrededor de 25 mm de grosor, mientras que el panel de la placa base suele ser más ancho y tiene más puertos).

Conecta el extremo USB del cable del visor a un puerto USB 3.0 (azul) del ordenador.

Conecta el cable del sensor a otro puerto USB 3.0 (azul) del ordenador.

A continuación, se activarán tres iconos verdes en la parte inferior izquierda de la pantalla de Oculus que indican que el visor y el sensor se han conectado correctamente.

Luego de conectar los periféricos dar clic en siguiente.

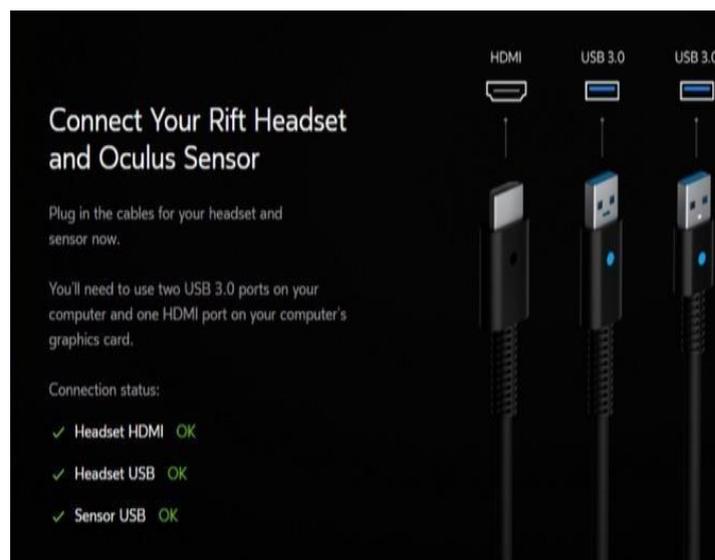


Ilustración 7: Configuración de los periféricos de Oculus conectados correctamente

NOTA: Si se visualiza iconos de advertencia rojos o amarillos o se experimenta algún otro problema, consultar en la página oficial de Oculus los posibles fallos y las posibles soluciones a los problemas de las gafas de realidad virtual.

Configuración sensores y mandos Oculus.

Para la configuración de los sensores y mandos de Oculus primero se tiene que habilitar los mandos de la realidad virtual para esto se debe colocar una pila doble A, en cada uno de los mandos. A continuación, se debe ubicar los sensores a los extremos del área de trabajo. (Visualizar la sección requisitos de trabajo)

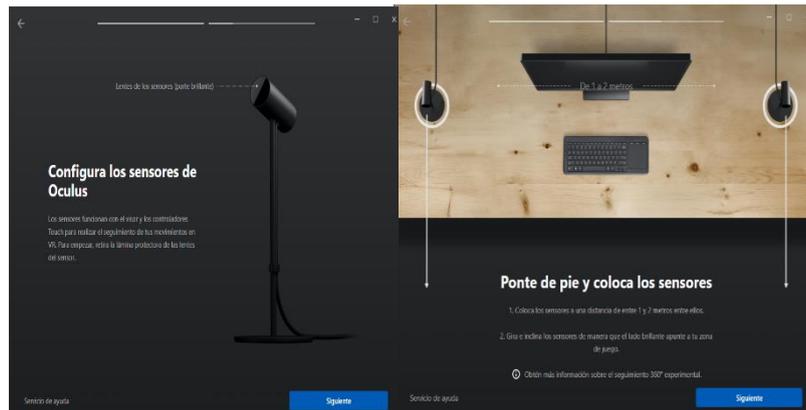


Ilustración 8: Visualización de pasos para ubicación de sensores

A continuación, una vez colocados los sensores se procede a mantener pulsado el botón “Seleccionar” situado en el centro del disco de navegación para emparejar el mando con el visor.

De esta forma se comprueba que efectivamente los sensores se encuentran en la ubicación correcta y que el mando se encuentra dentro del rango específico de la zona de trabajo.

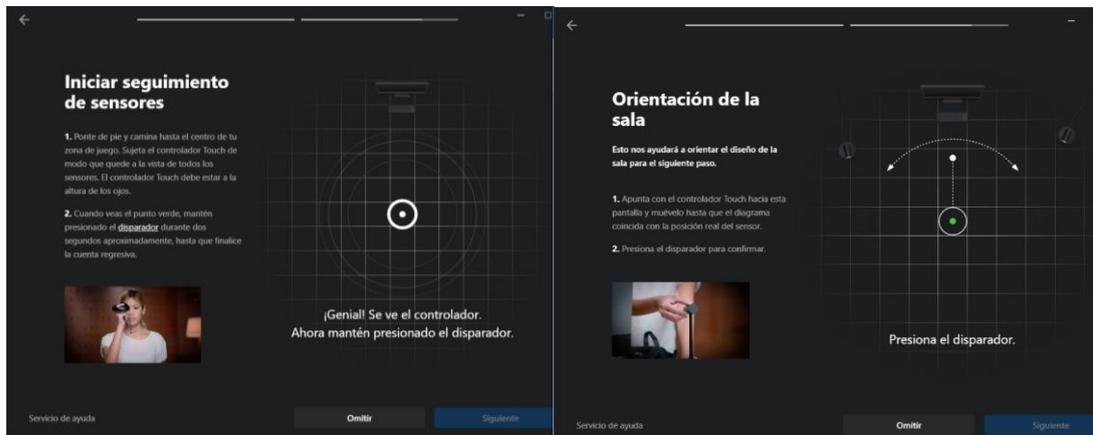


Ilustración 9: Configuración y ubicación de los sensores Oculus.

Cuando se ha configurado la ubicación adecuada de los sensores de realidad virtual, damos clic en siguiente para proceder a delimitar el área de trabajo.

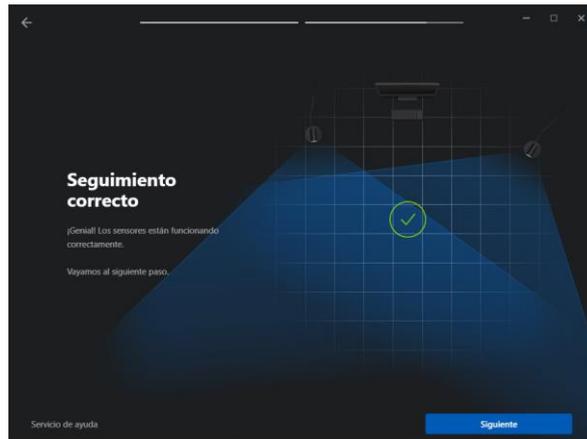


Ilustración 10: sensores y mandos de realidad virtual ubicados y configurados al 100%.

Configuración área de trabajo

Para el ara de trabajo se debe mantener pulsado el botón del disparador del lado derecho, y con el mismo se debe ir delimitando el perímetro que e desea usar para el manejo de la realidad virtual.

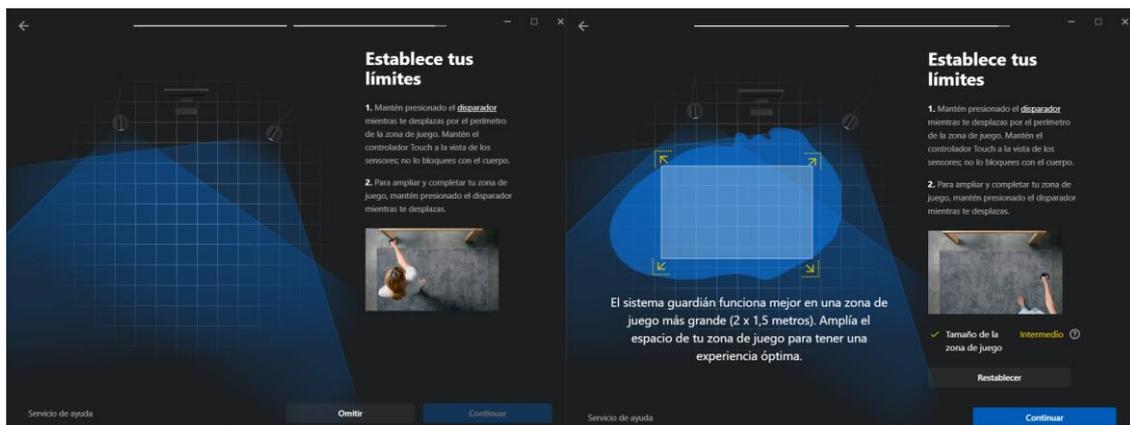


Ilustración 11: Delimitación del área de trabajo Oculus.

Tamaños del sistema guardián

El sistema guardián permite establecer límites en la realidad virtual que se hacen visibles si el usuario se acerca demasiado al perímetro de la zona de juego o control. Al configurar el sistema guardián, la zona de juego se mostrará con uno de estos tamaños:

Por debajo del mínimo: significa que la zona de juego tiene unas dimensiones inferiores a las mínimas establecidas de 1×1 metro para utilizar el sistema guardián.

Mínimo: tienes espacio suficiente para la zona de juego mínima necesaria, pero algunas aplicaciones pueden recomendar al menos 1.5×1.5 metros para disfrutar de la mejor experiencia.

Recomendado: tienes la zona de juego recomendada para las experiencias con desplazamiento.

Esto significa que tu zona de juego se ajusta al espacio de 2×2 metros que se necesita para configurar el sistema guardián.



Ilustración 12: Ejemplo del espacio del área de trabajo

Restablecer el sistema guardián

Una de las fallas más comunes en la parametrización del área de trabajo es una mala definición del sistema guardián, para esto existen ciertos pasos que se pueden usar para volver a configurar el área de trabajo.

Seleccionar Configuración en la barra de herramientas inferior.

Seleccionar Sistema guardián en el menú de la izquierda.

Hacer clic en Ajustar el sistema guardián.

Seguir las instrucciones que aparecen en pantalla para restablecer el sistema guardián.

Una vez configurada de forma efectiva el sistema guardián y la zona de trabajo una de las funciones más útiles de las gafas de realidad virtual es la comodidad que se puede adaptar a cada tipo de usuario que lo maneja, empezando por ajustar el nivel de sonido, posición de los auriculares, tamaño de y diámetro de la cabeza del usuario, focal de la vista del usuario, etc.

Plataforma de trabajo

Para el uso de la plataforma virtual inmersiva para el entrenamiento del control de procesos de flujo, se deberá conocer los entornos de trabajo, de los diferentes softwares que conforman la plataforma de entrenamiento.

LabView

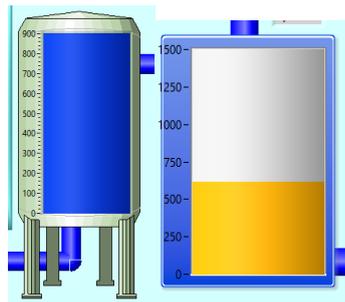
El primer entorno a reconocer es LabView, en este entorno se reconocerá las partes que lo conforman.



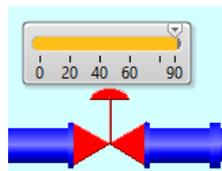
Ilustración 13: interfaz gráfica y visualizador de las variables del entorno virtual.

Dentro de las partes más importantes de la interfaz gráfica se encuentran:

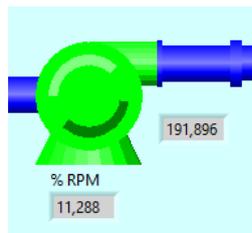
Tanques: se componen por los cilindros donde se puede visualizar el flujo de agua.



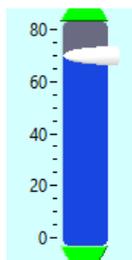
Válvulas: dispuestas a lo largo de la tubería para visualizar el control que existe desde Unity a través de realidad virtual.



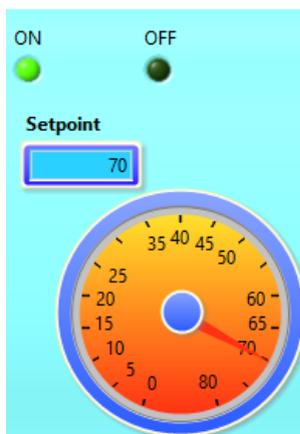
Bomba: visualiza el trabajo y el flujo que entrega a lo largo de la simulación



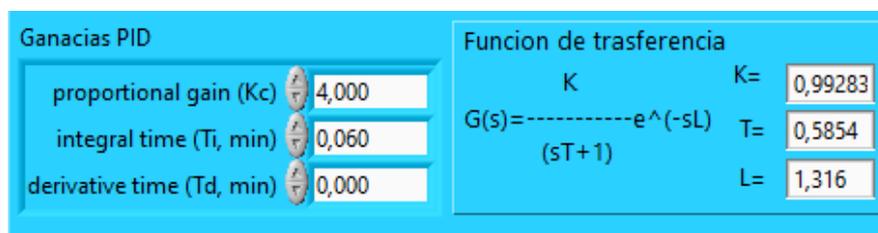
Flujómetro: visualizador del nivel de flujo manual que se indica a lo largo de la realidad virtual.



Parámetros de SetPoint: se dispone de un medidor circular o barómetro que indicara el valor al que se encuentra seteado el nivel de flujo que tiene que alcanzar la plataforma.



Valores de PID y función de transferencia: estos valores nos ayudan a visualizar los parámetros a los que e encuentra la plataforma.

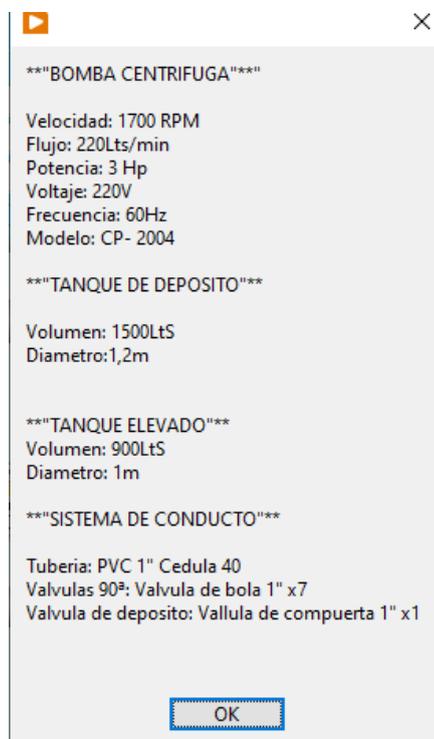


Conexiones entre sistemas y plataforma: se dispone de dos formas de conexión, la primera es una conexión TCP/IP la cual a través de la dirección IP a la que se encuentre el computador, así como el puerto a la que se realizó ping desde Unity, darán como resultado el enlace entre dichos programas.

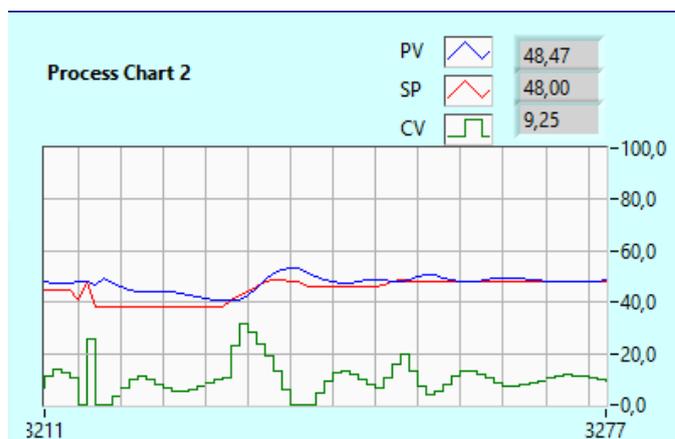
El otro modo de conexión es la comunicación serial la misma que se aplica entre LabView y Arduino para establecer el visualizador.



Especificaciones: será una pestaña de visualización de los parámetros en los que se encuentran definidos los datos del motor en uso.



Gráficos: se dispone de una ventana emergente donde se visualizará las curvas características del sistema.



Unity

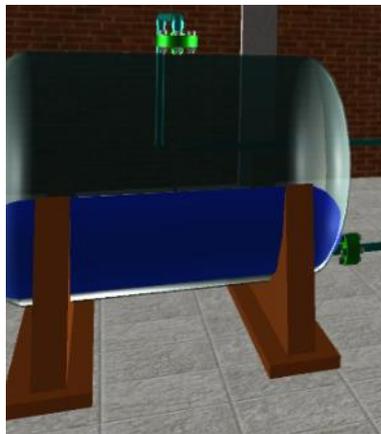
Dentro del entorno virtual de Unity se debe considerar las partes tridimensionales que conforman la planta.



Ilustración 14: Entorno grafico realizado en Unity

A continuación, se detallan las partes más importantes que conforman Unity.

Tanque de reserva: es el tanque de donde partirá el agua para alimentar a todo el sistema.

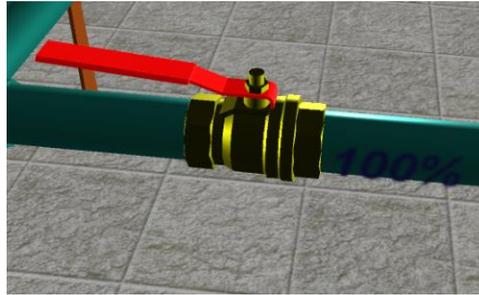


Bomba: dentro de la plataforma se encuentra la bomba que emula el real funcionamiento de la misma. Su función es el bombear el agua a través del sistema



Válvulas: son de media vuelta y su giro va de cero a noventa grados, aunque están escalados de

cero a cien en porcentaje.



Flujómetro manual: en la planta de flujo se colocó un medidor de flujo manual, el mismo que marca mediante elevación si existe flujo a través de la tubería.



Tablero de control: dentro de él se visualizará el nivel de SetPoint al que se desee llegar.



Flujómetro digital: dentro de la maqueta virtual existe un flujómetro virtual el mismo que marca de forma más exacta que el flujómetro manual.



Interfaz gráfica de Arduino

La interfaz gráfica de Arduino es un módulo real en donde se podrá visualizar parámetros de uso.

Visualizador gráfico (led verde), marca el funcionamiento del sistema

Visualizador gráfico (led rojo), marca el funcionamiento del motor

Visualizador gráfico (led rojo), mediante un pwm marca la forma de trabajo del sistema en la salida de forma real de un pin pwm de forma que en una futura actualización del sistema se le pueda ensamblar una bomba con pwm de forma real.

Display LCD en donde se visualizará el flujo actual en el que se encuentra actualmente la planta, así como el SetPoint establecido por el usuario.

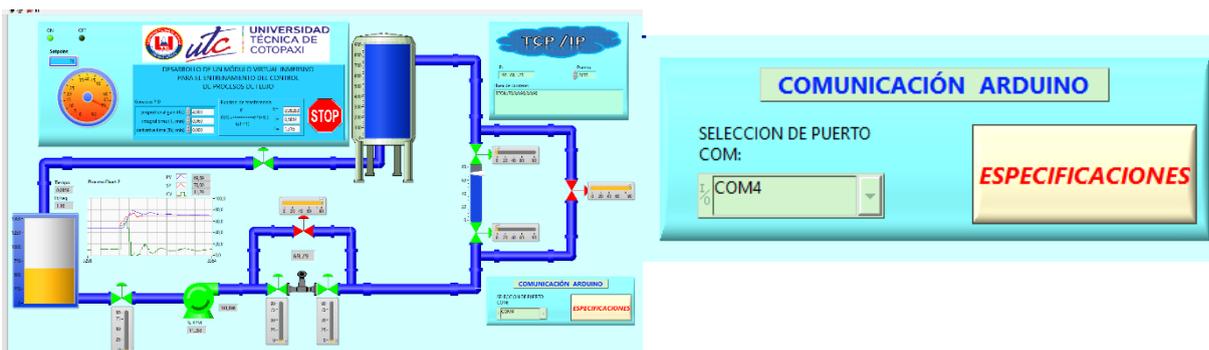


Ilustración 15: visualizador real de Arduino

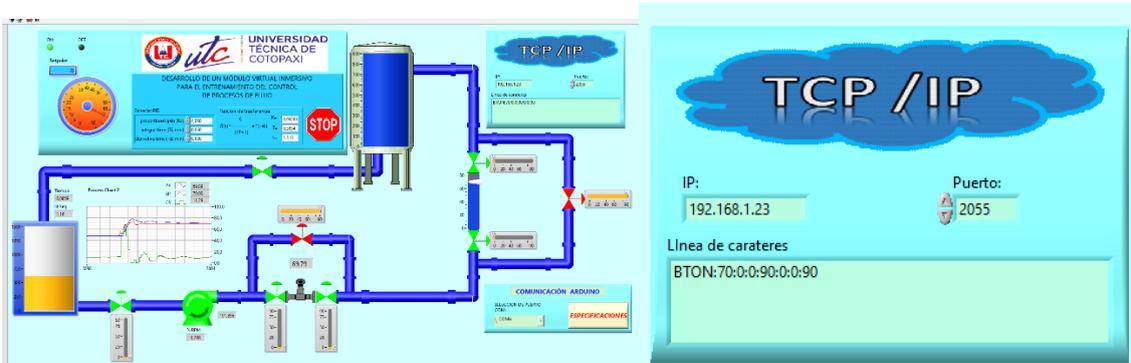
Puesta en marcha

Para la puesta en marcha de la plataforma virtual se debe seguir los siguientes pasos para un correcto funcionamiento.

- 1.- conectar y comprobar que funcione correctamente las gráficas de realidad virtual (Oculus) (como referencia consultar la sección del manual de uso)
- 2.- abrimos los programas que están inmersos con la plataforma virtual (LabView, Unity)
- 3.- dentro de la interfaz de LabView se escoge la comunicación serial en la misma se seleccionará el puerto COM, en el que se aloja el Arduino.



4.- a continuación, en la interfaz de LabView se ubica en la placa de comunicación TCP/IP y se llenan los datos pertinentes para la conexión de LabView con Unity. (como referencia se elige la dirección IP del computador a probar y el puerto de enlace del computador), para este caso se usa la dirección IP: 192.168.1.23 y el puerto de enlace será el PUERTO: 2055



5.- una vez llenos los datos de LabView, solo resta ubicarse en Unity y correr la simulación, a continuación, realizar lo mismo para LabView.

NOTA: cave recalcar que la interfaz con Arduino ya está precargada en el controlador por lo que no hace falta volver a cargar la programación en el mismo.

Podemos visualizar que Unity se encuentran en simulación cuando la misma interfaz cambia de color en su entorno.

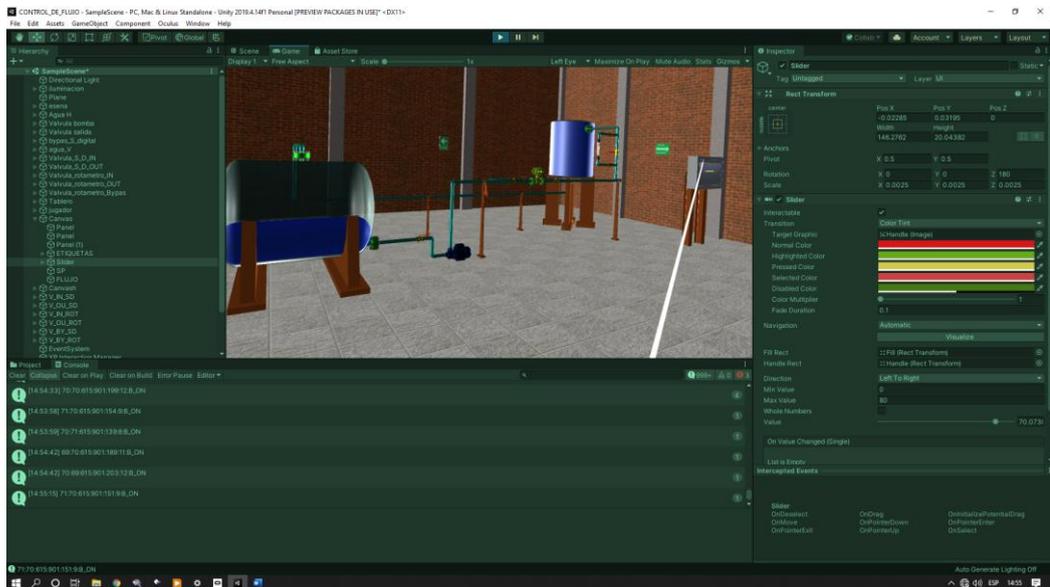


Ilustración 16: Cambio de color de la interfaz de Unity al momento de la simulación

También se puede visualizar que Arduino se encuentra ya en funcionamiento con el módulo de visualización comienza a marcar datos (ver ilustración 15).

Para la visualización de funcionamiento dentro de LabView se puede observar como la pantalla cambia de mostrar una grilla a ponerse en sólido, además de cambiar de color las flechas de

simulación de trabajo.

Se puede observar que se puede detener la simulación de LabView al presionar el botón de STOP.

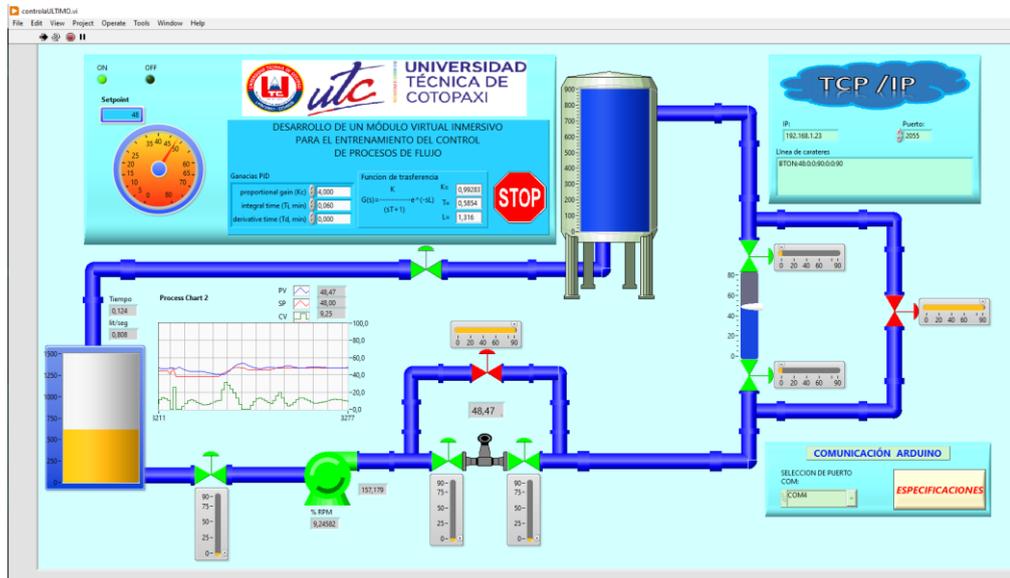


Ilustración 17: Programa e interfaz de LabView en funcionamiento

Manipulación

Para la manipulación del sistema o plataforma virtual se dispone de dos mandos otorgados por la realidad virtual, estos mandos son izquierdo y derecho. Estos mandos nos servirán para la movilización y manipulación de los objetos dentro de la plataforma virtual.

Con el control del lado derecho y el botón de disparo nos podremos mover hacia una zona específica, esto se logra apuntando la fecha de movimiento en el suelo de la plataforma virtual, de esta forma resultara más fácil viajar a través del entorno del proceso.

Con el control del lado izquierdo y el botón de disparo y sujeción se puede manipular las variables de trabajo dentro de la planta, tales como: SetPoint, Válvulas de paso y control, etc.



Ilustración 18: Mandos para la manipulación de realidad virtual