

**PROYECTO DE INVESTIGACIÓN SOFTWARE EDUCATIVO PARA EL ANALISIS
NO LINEAL DE PORTICOS 2D USANDO PROGRAMACION EN MATLAB**



Por:

Miguel Ángel Escobar García



**UNIVERSIDAD SANTO TOMÁS
FACULTAD DE INGENIERÍA CIVIL
VILLAVICENCIO
2021**

**PROYECTO DE INVESTIGACIÓN SOFTWARE EDUCATIVO PARA EL ANALISIS
NO LINEAL DE PORTICOS 2D USANDO PROGRAMACION EN MATLAB**



Por:

Miguel Ángel Escobar García

Documento final presentado como opción de grado para optar al título profesional de
ingeniero civil

Aprobado por:

Ing. Bregy Hassler Choque Jiménez, Msc.

Director

**UNIVERSIDAD SANTO TOMÁS
FACULTAD DE INGENIERÍA CIVIL
VILLAVICENCIO**

2021

Autoridades académicas

Fray José Gabriel Mesa Ángulo, O.P.

Rector General

Fray Eduardo González Gil, O.P.

Vicerrector Académico General

Fray José Antonio Balaguera Cepeda, O.P.

Rector Sede Villavicencio

Fray Rodrigo García Jara, O.P.

Vicerrector Académico Sede Villavicencio

Mg. Julieth Andrea Sierra Tobón

Secretaria de División Sede Villavicencio

Ing. Manuel Eduardo Herrera Pabón

Decano Facultad de Ingeniería Civil

Agradecimientos

Le quiero agradecer a Dios por permitirme vivir cada día, de haberme guiado, haber sido mi apoyo, luz y camino. Le agradezco a mi madre por ser el motor que me impulsa a seguir adelante cuando las adversidades se presentan en mis proyectos, por ser la guía en los momentos de confusión y la alegría en los días tristes. Agradecer a mis hermanos por el apoyo brindado a lo largo de mi vida.

Al Ingeniero Bregy Choque, por la confianza depositada en mi desde el primer día que se fundo la idea de este proyecto de grado, por sus consejos y enseñanzas que me ayudaron a lo largo de la carrera.

A la Universidad Santo Tomas, por cumplir con su misión de investigación y a través de sus herramientas físicas y digitales permitir el correcto y oportuno desarrollo de este proyecto de investigación.

Resumen

Este proyecto de investigación nace de la idea de ser un apoyo para los espacios académicos que corresponden al área de estructuras en la facultad de Ingeniería Civil, de manera que asista ambas partes del proceso de aprendizaje, afianzando las competencias que van adquiriendo los estudiantes y favoreciendo las labores del docente a través de la representación didáctica de los componentes prácticos.

Los cálculos estructurales que son necesarios para el análisis del comportamiento de un pórtico estructural siguen una metodología continua cuyas variables interactúan entre sí a lo largo de todo el procedimiento matemático, lo que supone una ventaja teórica que este software aprovecha de manera puntual, auxiliando las labores de los estudiantes y docentes de tal forma que todos los datos iniciales y los procesos que se ejecutan en el mismo son gráficamente suministrados para su correcto análisis y aprovechamiento. Este proyecto de investigación se figura establecer como la herramienta que asista al desarrollo de competencias teórico-prácticas en un método de análisis no lineal de pórticos en 2d como lo es la metodología de análisis de sismicidad PUSH-OVER, donde se analizan los comportamientos de desplazamientos y giros horizontales que presenta una estructura en la aparición de una fuerza horizontal externa.

Palabras Clave: *pórtico, estructura, software, Push-over, fuerza, sismicidad.*

Abstract

This research project was born from the idea of being a support for the academic spaces that correspond to the area of structures in the Faculty of Civil Engineering, in a way that supports both parts of the learning process, strengthening the skills that students are acquiring and favoring the work of the teacher through the didactic representation of the practical components.

The structural calculations that are necessary for the analysis of the behavior of a structural gantry follow a continuous methodology whose variables interact with each other throughout the entire mathematical procedure, which represents a theoretical advantage that this software takes advantage of in a specific way, helping the tasks of the students and teachers in such a way that all the initial data and the processes that are executed in it are graphically supplied for their correct analysis and use. This research project is established as the tool that assists in the development of theoretical-practical skills in a non-linear analysis method of 2d frames such as the PUSH-OVER seismicity analysis methodology, where displacement behaviors are analyzed and horizontal turns that a structure presents in the appearance of an external horizontal force.

Key Word: gantry, structure, software, pushover, force, seismicity.

Contenido

Formulación Del Problema.....	15
Descripción Del Problema.....	15
Formulación Del Problema.....	15
Justificación	16
Objetivos.....	17
Objetivo General.....	17
Objetivos Específicos.....	17
Marco de Referencia.....	18
Marco Teórico	18
Modelo o formula de Hognestad	18
Modelo de Kent y Park.....	18
Aplicación del Push-over.....	20
Marco Conceptual.....	25
Antecedentes	26
Metodología	27
Planteamiento Del Problema Ejemplo	27
Etapas 1	27
Lenguaje de programación en MatLAB	27
Acceso a AppDesigner.....	27
Elementos de programación	30

Etapa Dos	34
Cálculo del momento curvatura.....	34
Etapa Tres	37
Selección De Pórtico	37
Cálculo De Momento Curvatura	38
Calculo Grafico Cortante Basal-Desplazamiento.....	39
Problema Base	40
Resultados.....	41
Selección De Pórtico.....	41
Interfaz.....	41
Calculo Momento-Curvatura	41
Interfaz.....	42
Calculo Cortante Basal-Desplazamiento	42
Interfaz Primer Nivel.....	43
Interfaz segundo nivel	43
Interfaz tercer nivel	44
Análisis De Resultados	45
Cálculo De Momento Curvatura	45
Calculo Cortante Basal-Desplazamiento	46
Resultados Pórtico De Un Nivel	47
Resultados Pórtico Dos Niveles	49
Resultados Pórtico Tres Niveles	51
Graficas Obtenidas	55

Momento Curvatura	55
Cortante Basal Desplazamiento Pórtico Dos Niveles	55
Cortante Basal-Desplazamiento Pórtico Tres Niveles	56
Conclusiones Y Trabajos Futuros	58
Conclusiones	58
Trabajos Futuros.....	58
Bibliografía.....	59
Anexos.....	61
Código De Programación Selección de pórtico	61
Código De Programación Calculo momento curvatura.....	61
Cortante Basal-Desplazamiento.....	64
Código De Programación Un Nivel.....	64
Código Programación Dos Niveles.....	67
Código De Programación Tres Niveles	69

Lista De Tablas

Tabla 1. Datos y Calculos pseudoaceleracion.....	47
Tabla 2. Datos y calculos cortante basl y fuerza horizontal.	47
Tabla 3. Analisis a columnas primer piso	48
Tabla 4. Pseudoaceleracion.....	49
Tabla 5. Cortante basal y fuerza horizontal.	49
Tabla 6. Analisis a columnas piso 1.	50
Tabla 7. Análisis a columnas piso 2	50
Tabla 8. Cálculos pseudoaceleracion.....	51
Tabla 9. Analisis a coliumnas del piso 1.....	52
Tabla 10. Analisis a columnas piso 2.	52
Tabla 11. Analisis a columnas piso 3	53
Tabla 12. Cortante basal y fuerzas horizontales.....	53
Tabla 13. Derivas del análisis de dos pisos.....	56
Tabla 14. Derivas del analisis de tres pisos.....	56

Lista de Figuras

Figura 1. Modelo esfuerzos-deformaciones	18
Figura 2. Modelo Kent y Park	19
Figura 3. Comportamientos analizados modelo Kent y Park.....	20
Figura 4. Probabilidad de excedencia.....	21
Figura 5. Nivel de trabajo ATC-40.....	22
Figura 6. Modelo curva capacidad.	22
Figura 7. Análisis Push-over, espectro sísmico.....	23
Figura 8. Análisis sísmico modal.....	23
Figura 9. Modelo demanda Push-over.....	24
Figura 10. Punto de trabajo de la estructura.	24
Figura 11. Amortiguamiento efectivo.....	25
Figura 12. Icono de escritorio MatLAB.	28
Figura 13. Interfaz MatLAB.	28
Figura 14 Comando para acceder a AppDesigner.	29
Figura 15. Interfaz de creación AppDesigner.	29
Figura 16. Appdesigner en Blank Mode.	30
Figura 17. Code View AppDesigner.....	30
Figura 18. Orden Callback en Design View.....	31
Figura 19. Orden Callback en Code View.	32
Figura 20. Ejemplo calculo área en Design view.....	33
Figura 21. Code View del programa área.	34
Figura 22. Algoritmo selección pórtico.	38
Figura 23. Algoritmo Momento Curvatura.	39
Figura 24. algoritmo cortante basal-desplazamiento.....	40
Figura 25. Interfaz selección de pórtico.	41
Figura 26. Interfaz calculo momento-curvatura.	42

Figura 27. Interfaz cortante basal-desplazamiento pórtico un nivel.....	43
Figura 28. Calculo cortante basal-desplazamiento dos niveles.....	43
Figura 29. Calculo cortante basal-desplazamiento tres niveles.....	44
Figura 30. Comprobación resultados momento-curvatura.....	46
Figura 31. Cálculos primer nivel cortante basal-desplazamiento.....	48
Figura 32 Calculo y resultados cortantes basal-desplazamiento.....	51
Figura 33. Calculo cortante basal y desplazamiento.	54
Figura 34. Calculo cortante basal y desplazamiento.	54
Figura 35. Grafica momento curvatura.....	55
Figura 36 Cortante-Basal desplazamiento dos niveles.	55
Figura 37 Cortante basal-desplazamiento tres niveles.....	56

Lista De Ecuaciones

Ecuación 1 Tramo A.....	19
Ecuación 2 Tramo B.....	19
Ecuación 3 Tramo C.....	20
Ecuación 4. Método capacidad-demanda.....	21
Ecuación 5. Sa	22
Ecuación 6 Inercia.....	34
Ecuación 7 Modulo de elasticidad	34
Ecuación 8. Momento fisurante.	35
Ecuación 9. Φ fisurante	35
Ecuación 10. Valor n.	35
Ecuación 11. Area del elemento.....	35
Ecuación 12. Altura al centro del acero de refuerzo	35
Ecuación 13 Cuantia.	35
Ecuación 14. Valor x.	35
Ecuación 15. Gs.....	36
Ecuación 16. Módulo de elasticidad en límite de servicio.	36
Ecuación 17. • Φ límite de servicio.....	36
Ecuación 18. Gs limite de servicio	36
Ecuación 19. Momento uno prevención colapso.....	36
Ecuación 20. Módulo de elasticidad limite operativo.....	36
Ecuación 21. • Φ uno en prevencion colapso.....	36
Ecuación 22. Valor x2.	37
Ecuación 23. Momento dos en limite colapso.....	37
Ecuación 24. • Φ dos en limite colapso.....	37
Ecuación 25. • Φ de colapso.....	37

Titulo

Software educativo para el análisis no lineal de pórticos 2d usando programación en MatLAB

La creación de un software educativo dirigido a complementar la educación del estudiante de ingeniería civil o campos afines a través de la aplicación del análisis sísmico no lineal PUSHOVER es el fin con el que se planteó la elaboración de este proyecto de investigación. Con esta herramienta el estudiante puede reconocer el estado actual de sus competencias y habilidades en temáticas relacionadas al análisis estructural y concreto reforzado al solucionar problemas que tomen en cuenta el comportamiento sísmico de una estructura.

Con esta herramienta educativa se pretende garantizar que el estudiante pueda desenvolverse en situaciones teóricas con grados de dificultad medio y alto para reconocer de manera didáctica y exitosa la curva de capacidad de una estructura, en este caso, cualquier pórtico estructural simple en 2d.

La programación en MatLAB fue el primer paso a tomar para construir la gama de ventajas que se pretende entregar al estudiante para apoyar el desarrollo de sus capacidades y competencias de análisis estructural, debido a que la suficiencia del alumno para la comprensión de los resultados obtenidos inicia en el código interno del software, programado en Lenguaje M seleccionado debido a su semejanza de escritura al usado en Visual Basic Advance, lenguaje de programación que es impartido en el espacio académico de lenguaje y programación.

El alcance de este proyecto de investigación de pregrado se encuentra en la construcción del software didáctico en MatLAB para la construcción y obtención de variables correspondientes a las gráficas Momento-Curvatura, Cortante Basal-Desplazamiento y Fuerza horizontal-Desplazamiento correspondientes a la metodología Push-over de un pórtico simple en 2D.

Las limitaciones encontradas en el desarrollo del proyecto fueron las que ofrece la interfaz gráfica del AppDesigner al impedir la construcción de una sola ventana correspondiente al desarrollo del ejercicio planteado, lo que fue resuelto a través de la creación de una multiventana intercomunicada con variables conectadas a través de la herramienta.

Formulación Del Problema

Descripción Del Problema

De acuerdo a Pappalardo en su obra “Enseñanza tradicional: características, metodología, ventajas y desventajas”, la principal desventaja de la metodología que se sigue en la academia para impartir cualquier temática es la poca o nula adaptabilidad que puede proporcionar el docente a sus alumnos individualmente debido a la cantidad que posee por aula, esto impide que los ritmos de aprendizaje se puedan balancear ignorando totalmente la implementación de una estrategia de aprendizaje individual debido a que cada alumno aprende a una proporción distinta que sus pares. (Pappalardo, 2017)

La educación tradicional también impide que los estudiantes participen en el método de enseñanza limitándolos a cumplir únicamente el papel de receptores de información y al docente como ente comunicador de la información de una manera teórica y en rasgos generales para dejar que las particularidades de la temática sean adquiridas por los estudiantes de manera autodidacta.

Según la Lic. Solana, la tecnología educativa es una herramienta muy polémica en la actualidad, los centros educativos evitan implementarla en su totalidad debido a que falsamente se cree que se implementa con el motivo de disminuir el talento humano en las actividades de enseñanza. (Solana, Arias, & Gomez, 2012)

El software educativo no es una amenaza para el docente, es un producto diseñado para apoyarlo, disminuirle la carga laboral que se produce por la densidad de alumnos presente en un aula y mejorar las metodologías educativas permitiendo que pueda acercarlos a la realidad de lo que se enseña y que interactúen con esa información de tal manera que todo conocimiento pueda retroalimentarse individualmente y evaluarse de tal manera que las habilidades y competencias de cada estudiante puedan desarrollarse conforme la temática evolucione y el curso avance.

Formulación Del Problema

¿De qué manera el uso de un software educativo ayuda al aprendizaje del comportamiento sísmico de un pórtico simple de concreto armado en 2d en los estudiantes de asignaturas afines al análisis estructural?

Justificación

El desarrollo de este proyecto de investigación se basa en satisfacer la necesidad académica de promover el desarrollo de herramientas de programación que ayuden a la enseñanza y aprendizaje del comportamiento sísmico de las infraestructuras y el cómo funcionan los análisis estructurales a través de metodologías de resolución de ecuaciones diferenciales no lineales.

Los estudiantes necesitan de un apoyo académico didáctico y realista que suponga darles un vistazo a lo que tendrán en su futuro profesional como cotidianidad, los sistemas constructivos tienen como factor común las estructuras de pórticos, cuyo comportamiento se rige a través de la unión de columnas y vigas gobernadas a flexión. De tal manera que un método de cálculo como el que se propone en este proyecto de investigación es la unión de diferentes disciplinas impartidas normalmente en la carrera de ingeniería civil y simular de forma didáctica el comportamiento físico de una infraestructura con respecto a los movimientos sísmicos que pueda presentar a lo largo de su ciclo de vida.

Objetivos

Objetivo General

Desarrollar un aplicativo didáctico computacional en el lenguaje M de MatLAB que permita la obtención de la curva capacidad de un pórtico simple en concreto armado de hasta tres niveles.

Objetivos Específicos

- Elaborar los diagramas de flujo de los algoritmos base para la programación de los análisis Push-over de los pórticos simples de uno, dos y tres niveles.
- Describir el procedimiento teórico para la obtención de las curvas momento-curvatura de elementos estructurales en concreto armado.
- Exponer la metodología Push-over para el análisis no lineal para la obtención de la curva de capacidad de una estructura.
- Diseñar el programa de cálculo de análisis no lineal Push-over con el lenguaje M del software MatLAB.
- Realizar la validación de los resultados a través de la teoría aportada por la literatura especializada en el método Push-over.

Marco de Referencia

Marco Teórico

La metodología Push-over es el análisis estático no lineal que consiste en determinar la curva de capacidad de una estructura a través de un patrón de cargas dado.

Modelo o formula de Hognestad

Donde f_c y ϵ_c son el esfuerzo y la deformación unitaria en el concreto y f'_c y ϵ_{cu} son la deformación unitaria máxima útil del concreto en compresión, este modelo está compuesto por una parte curvilínea descrita por una parábola que en f'_c describe la resistencia máxima alcanzada por el concreto y la parábola es tangente a al horizontal.

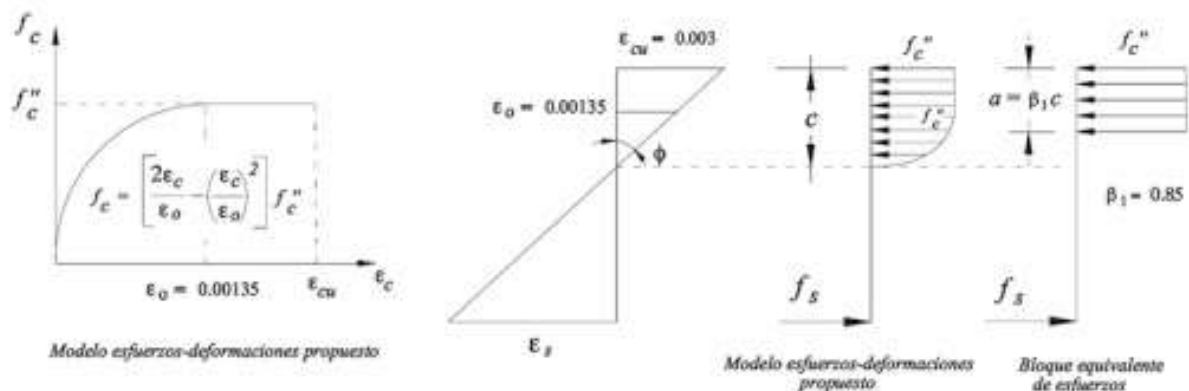
Para $0 < \epsilon_c < \epsilon_0$

$$f_c = f'_c \left[2\epsilon_c / \epsilon_0 - (\epsilon_c / \epsilon_0)^2 \right]$$

Para $\epsilon_0 < \epsilon_c < \epsilon_{cu}$

$$f_c = f'_c - 0.15 \cdot f'_c (\epsilon_c - \epsilon_0) / (\epsilon_{cu} - \epsilon_0)$$

Figura 1. Modelo esfuerzos-deformaciones



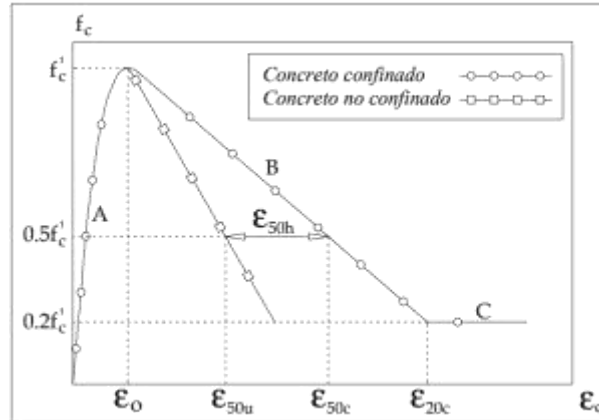
Nota: Adaptado de (Gutierrez, 2015)

Modelo de Kent y Park

Este modelo es únicamente aplicable para modelos de secciones rectangulares o cuadradas, el cual considera que el confinamiento no tiene algún efecto sobre la resistencia

igualándolo a un concreto normal. Este modelo se basa en la curva mostrada en la figura a continuación:

Figura 2. Modelo Kent y Park



Nota: Adaptado de (Gutierrez, 2015)

Este modelo este compuesto por 3 tramos que forman una curva y los tramos de depender de las siguientes condiciones:

Tramo A: En este primer momento aun no presenta confinamiento y tiene el comportamiento de un cemento normal, esta es una parábola en segundo grado y se rige bajo el siguiente intervalo y ecuación.

$$f_{cA} = f'_c \left[\frac{2\varepsilon_c}{\varepsilon_0} - \left(\frac{\varepsilon_c}{\varepsilon_0} \right)^2 \right] \rightarrow 0 \leq \varepsilon_c \leq \varepsilon_0$$

Ecuación 1 Tramo A

Tramo B: Esta corresponde a una ecuación lineal, la cual comienza cuando el concreto ya obtuvo su máxima resistencia y finaliza cuando su resistencia se ha disminuido un 80% y su ecuación es la siguiente.

$$f_{cB} = f'_c [1 - Z(\varepsilon_c - \varepsilon_0)] \rightarrow \varepsilon_0 \leq \varepsilon_c \leq \varepsilon_{20c}$$

Ecuación 2 Tramo B

Tramo C : Esta muestra que el concreto puede tomar la resistencia de 20 % del total como máximo y no tendrá esfuerzos adicionales y la ecuación es la siguiente.

$$f_c = 0.2 f'_c \rightarrow \varepsilon_c > \varepsilon_{20c}$$

$$Z = \frac{1}{2(\varepsilon_{50u} + \varepsilon_{50h} - \varepsilon_0)}$$

$$\varepsilon_{50u} = \frac{3 + \varepsilon_0 f'_c}{f'_c - 1000}$$

$$\varepsilon_{50u} = \frac{3}{4} \rho_s \sqrt{\frac{bc}{s}}$$

Ecuación 3 Tramo C

Donde:

ε_c = Deformación unitaria del concreto

ε_0 = Deformación unitaria asociada a la resistencia máxima a compresión del concreto f'_c .

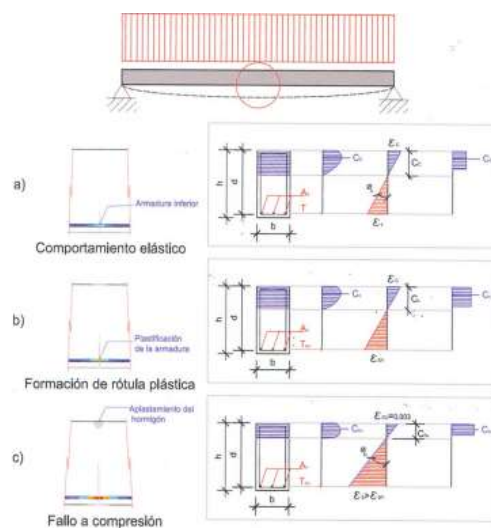
ε_{20c} = Deformación al 20%

Bc = Ancho de la sección.

S = separación de los estribos

ρ_s = Relación entre el volumen de los estribos y el concreto de confinamiento.

Figura 3. Comportamientos analizados modelo Kent y Park.



Nota: Adaptado de (Gutierrez, 2015)

Aplicación del Push-over

Para después de un sismo, determinar el estado final de la infraestructura es una situación de trato complejo por lo tanto el método de Push-over por capacidad-demanda permite una

sustitución de la estructura por un sistema equivalente en forma modal no lineal describiéndola por propiedades modales efectivas.

El Push-over tiene como bases teóricas las siguientes normas:

- FEMA (1997) La norma guía para la rehabilitación sísmica de edificaciones por la Federal Emergency Management Agency de Washington.
- ATC-40 (1996) La evaluación sísmica y modernización de edificaciones en concreto del Concejo de Tecnologías Aplicadas de California.

La capacidad de la estructura se determina a través del método Push-over al simular una distribución de carga parecida a la dominante en la estructura real, obteniendo la demanda del espectro elástico de diseño de la estructura, donde a través de un proceso iterativo se reconoce que la demanda y la capacidad en un evento sísmico no son independientes. Este método es óptimo ante estructuras cuya respuesta ante sismos es dominada por un modo de vibración.

$$T = -t/LN(L - P)$$

Ecuación 4. Método capacidad-demanda

T = Periodo de retorno

t = Tiempo de exposición

p = Probabilidad de excedencia

Figura 4. Probabilidad de excedencia

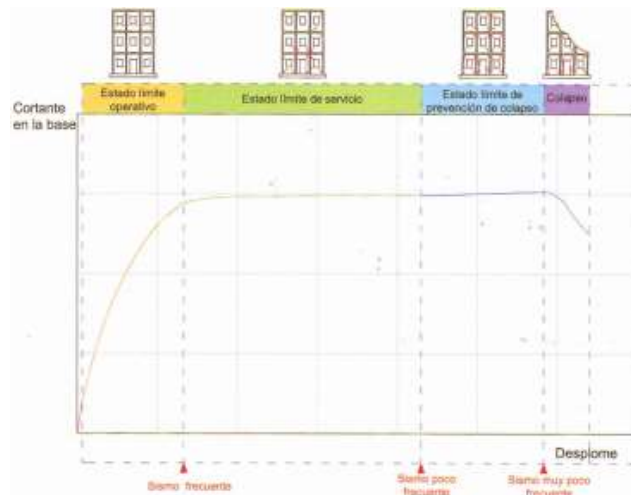
Nivel de sismo	T (años)	Probabilidad de excedencia (p,t)
Frecuente (S.Servicio)	43	50 % en 30 años
Raro (S.Diseño)	475	10 % en 50 años
Muy Raro (S. Máxima)	970	10 % en 100 años

Nota: Adaptado de (Gutierrez, 2015)

Figura 5. Nivel de trabajo ATC-40

	Nivel de trabajo de la edificación, criterio de seguridad (ATC-40) para edificaciones convencionales			
Nivel de sismo	Operacional	Inmediata Ocupación	Seguridad Vital	Estabilidad estructural
SS				
SD			X	
SM				X

Nota: Adaptado de (Gutierrez, 2015)

Figura 6. Modelo curva capacidad.

Nota: Adaptado de (Gutierrez, 2015)

Transformación de la curva de capacidad al espacio ADSR (Espectro de pseudoaceleración S_a – espectros de desplazamiento relativos S_d)

Asumiendo que el modo que controla la respuesta es el fundamental:

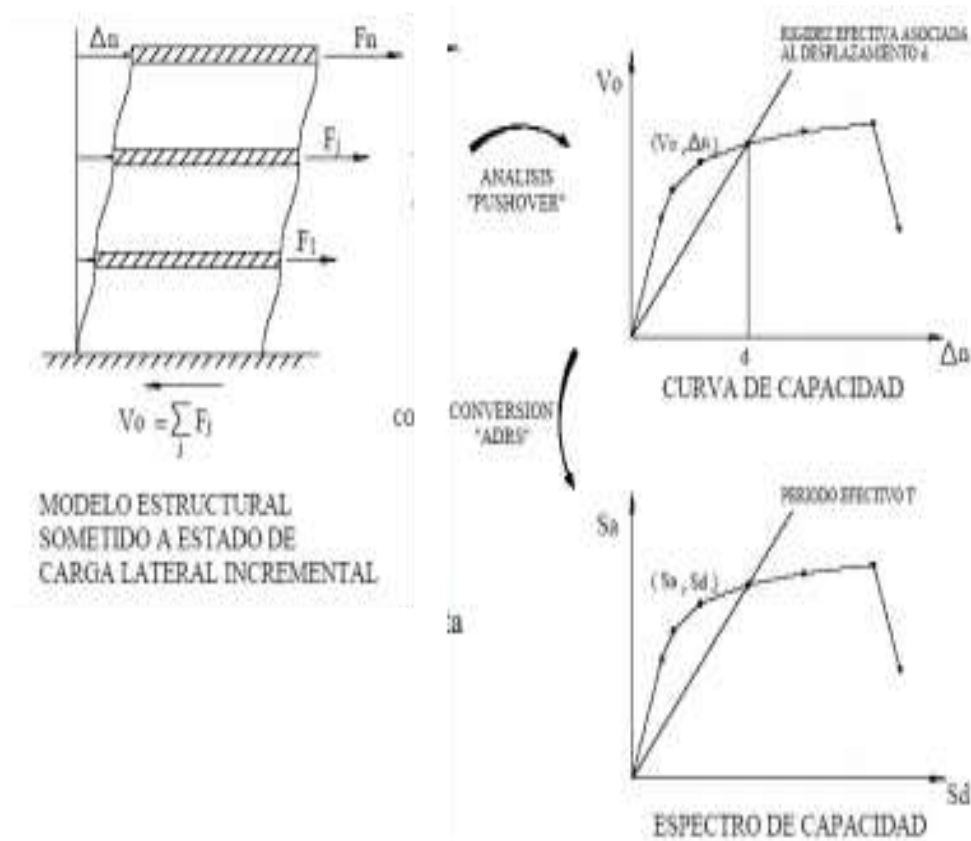
$$S_{ai} = \frac{V_i}{M_1} = \frac{V_i}{\phi_{m1} \phi_{1n}}$$

Ecuación 5. S_a

M_1 =Masa asociada al modo1

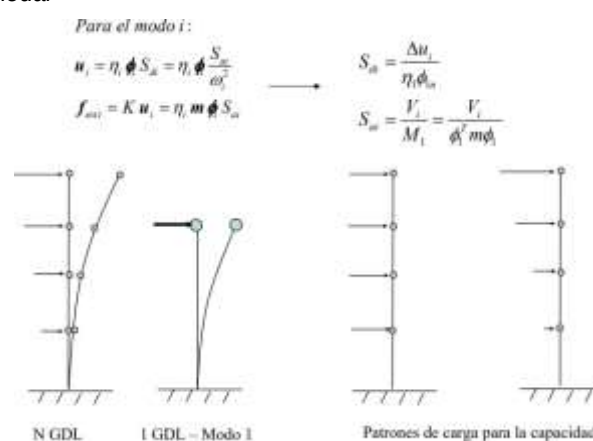
ϕ_{1n} = Amplitud máxima del modo de vibración 1

n_1 = Factor de participación de modo 1

Figura 7. Análisis Push-over, espectro sísmico.

Nota: Adaptado de (Gutierrez, 2015)

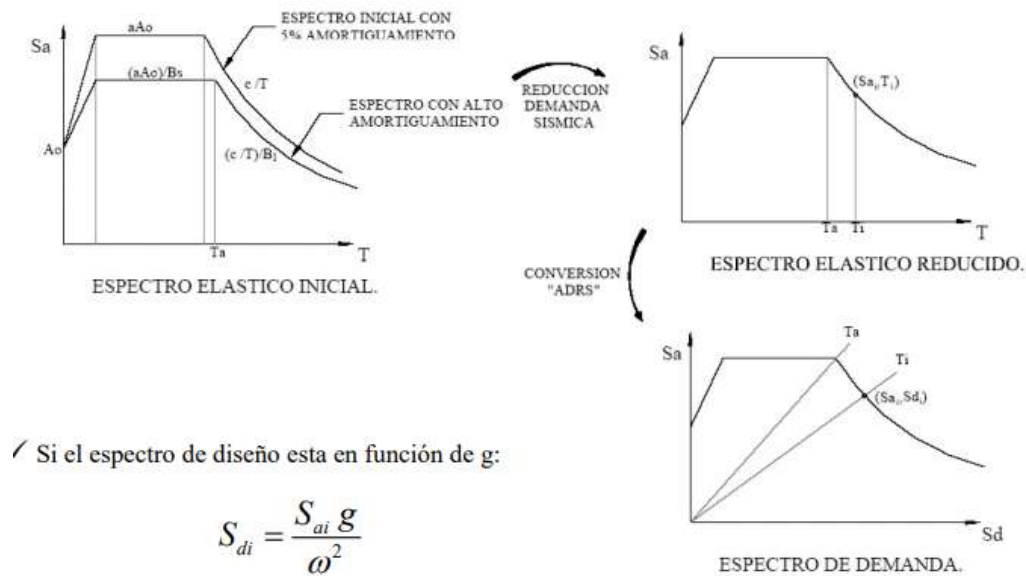
Se trata de aplicar las fórmulas para el análisis sísmico modal:

Figura 8. Análisis sísmico modal

Nota: Adaptado de (Gutierrez, 2015)

Espectro de demanda: Representación gráfica de la demanda sísmica que presenta la respuesta máxima de sistemas de un grado de libertad en función de sus frecuencias. (Díaz, 2016)

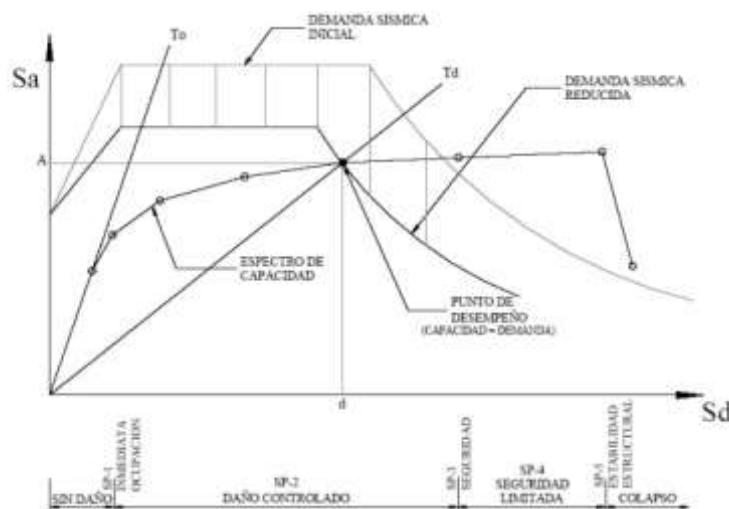
Figura 9. Modelo demanda Push-over.



Nota: Adaptado de (Gutierrez, 2015)

Punto de trabajo de la estructura: Es la representación del punto máximo de desplazamiento que se espera sufra una estructura ante una demanda sísmica determinada. (Bohorquez, 2016)

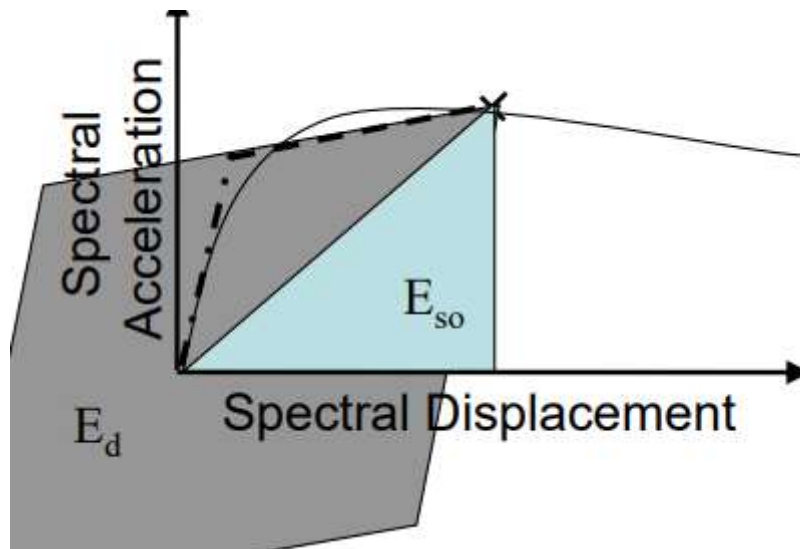
Figura 10. Punto de trabajo de la estructura.



Nota: Adaptado de (Gutierrez, 2015)

Amortiguamiento efectivo: Es la disipación de energía de una determinada estructura por acción de una fuerza dinámica proporcional a la velocidad del elemento o a los desplazamientos de un comportamiento no lineal físico (Moller, Rubinstein, & Ascheri, 2011)

Figura 11. Amortiguamiento efectivo.



Nota: Adaptado de (Gutierrez, 2015)

Marco Conceptual

Un pórtico simple es un armazón o estructura que se compone del conjunto de dos columnas conectadas entre sí por una viga superior, puede ser elaborado en distintos materiales, pero tradicionalmente en concreto armado. Esta estructura permite a través de sistemas aporticados ofrecerle a la estructura en la que se implementen ciertas ventajas como la flexibilidad en la distribución de los espacios, atraen pequeños requerimientos sísmicos y la ductilidad que poseen sus elementos le permiten disipar la energía al ofrecer una gran hiperestaticidad.

MatLAB es la abreviatura de Matrix Laboratory, lo que traducido al español sería laboratorio matricial o de matrices. Es un software matemático lanzado al público el año de 1984 por MathWorks, también creadores de SIMULINK, cuya población mercantil objetivo eran los centros educativos y de investigación al ofrecerles una herramienta que permitiera la manipulación de matrices y con el paso del tiempo se le fue incluyendo la posibilidad de la inserción de algoritmos y creación de UI's.

Antecedentes

El método Push-over es un método de análisis no lineal de la respuesta de los edificios ante la aparición de una fuerza horizontal que consiste en el desplazamiento producido por esta hasta llegar al colapso.

El ingeniero Ramón Manuel Calcina Peña en el desarrollo de su tesis de maestría realizó la evaluación del desempeño sísmico de un edificio de once pisos a través de la metodología de análisis estático y dinámico no lineal determinando gracias a esta que la ductilidad lateral se incrementaba inversamente a la cantidad de la cuantía del acero presente en las columnas.

El estudiante Jairo David Valencia determinó que una estructura en hormigón armado, en el cantón de Ambato, se encontraba en el rango de colapso preventivo a través del análisis modal Push-over en el desarrollo de su tesis de pregrado.

Los estudiantes Cesar Duarte, Marlon Martínez y José Santamaría a través del método Push-over determinaron que el edificio de la facultad de medicina de la universidad del Salvador poseía elementos en situación de deterioro y medio graves a espera de reparaciones para garantizar la seguridad de la estructura y sus ocupantes, en el desarrollo de su tesis de pregrado.

Los documentos anteriormente mencionados son la base teórica suficiente para la construcción del algoritmo que será la guía para la programación del software educativo que se supone desarrollar a lo largo de este proyecto de investigación.

Complementando lo previamente expuesto, la aplicación del software MatLAB como instrumento de enseñanza de matemática I en los estudiantes del I ciclo de la carrera de Ingeniería de sistemas de la Universidad de Ciencias y humanidades en Lima realizada por el estudiante Efracio Asis para la obtención de su grado Magister y el desarrollo de una interfaz de programación en MatLAB para la simulación de tareas de control sobre un kernel de tiempo real del estudiante Roberto Linares para la obtención de su Maestría en Ciencias son los documentos teóricos esenciales para reconocer el apoyo que supone para un estudiante tener a su alcance una herramienta digital que complemente su educación y la manera en que se programan estos mismos en la interfaz de escritura de MatLAB.

Metodología

Planteamiento Del Problema Ejemplo

Para iniciar con la realización del programa se plantea inicialmente el problema ejemplo para reconocer si la programación arroja los valores y gráficos correctos.

Se plantea un pórtico simple en 2d, compuesto por dos columnas y una viga de 40cm de base y 60 cm de altura con una estructura de acero compuesta por 4 varillas #6.

Para lograr la correcta y completa ejecución de este proyecto de investigación se plantea construir una metodología cronológica dividida en un numero de etapas con el fin de cumplir los objetivos específicos propuestos para la misma:

Etapas 1

Lenguaje de programación en MatLAB

A través de las teorías extraídas en el proceso de la construcción del estado del arte y el manual de usuario que presenta MatLAB en su sitio web, se estudiaron todas las formas en que se comporta el lenguaje M de MatLAB en el desarrollo de aplicaciones graficas “*Respond User*” que se desarrollan a través del software de AppDesigner.

Acceso a AppDesigner

AppDesigner es el entorno gráfico y programable que ofrece MatLAB para que sus usuarios desarrollen aplicaciones, que ellos denominan “*Respond User*”, a través del lenguaje M y definir su comportamiento.

Para acceder a este entorno de desarrollo se debe acceder a la aplicación de escritorio de MatLAB:

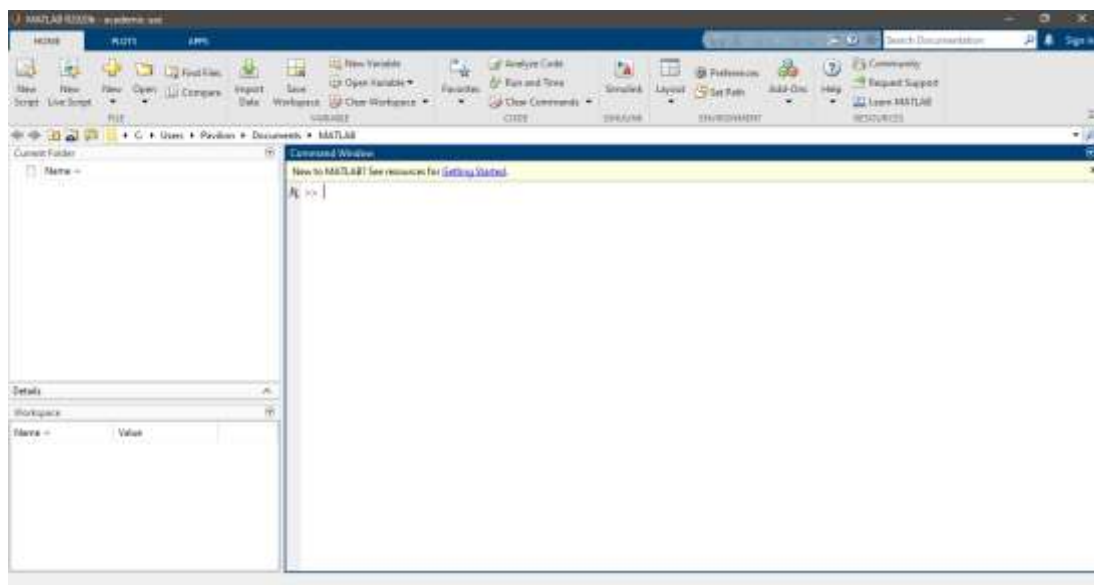
Figura 12. Icono de escritorio MatLAB.



Nota: Por Miguel Escobar

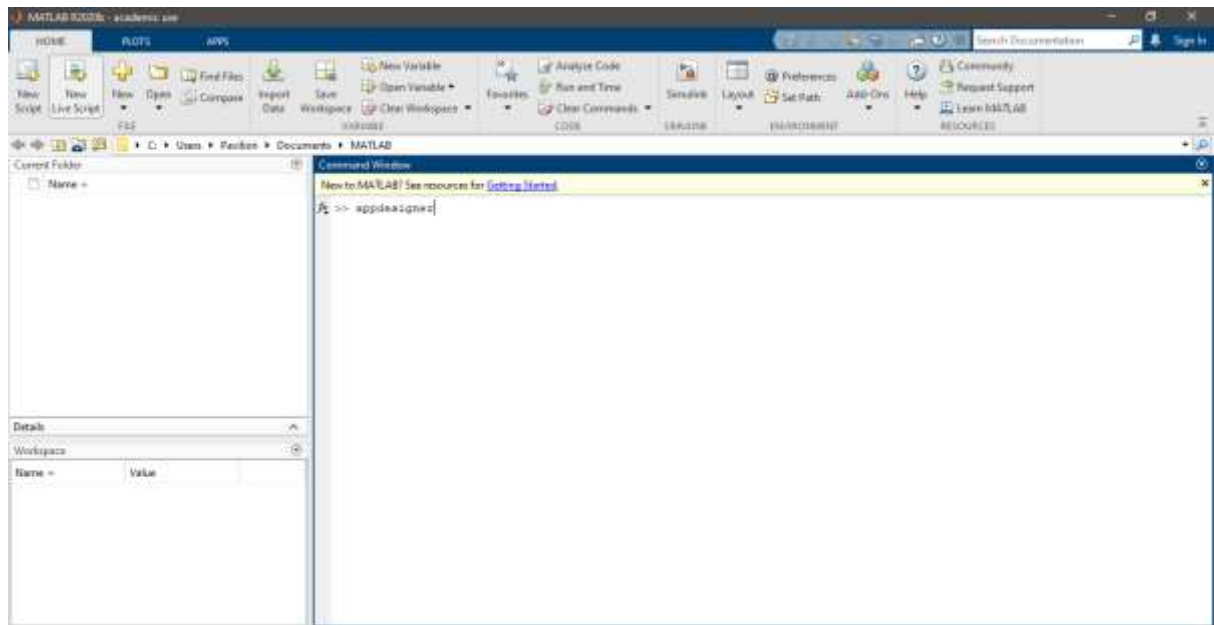
El software es abierto y se despliega la interfaz *blank Project* en la cual se nos muestra la *Command Window* que es el puente entre la interfaz de MatLAB y sus demás interfaces hermanas como lo es AppDesigner.

Figura 13. Interfaz MatLAB.



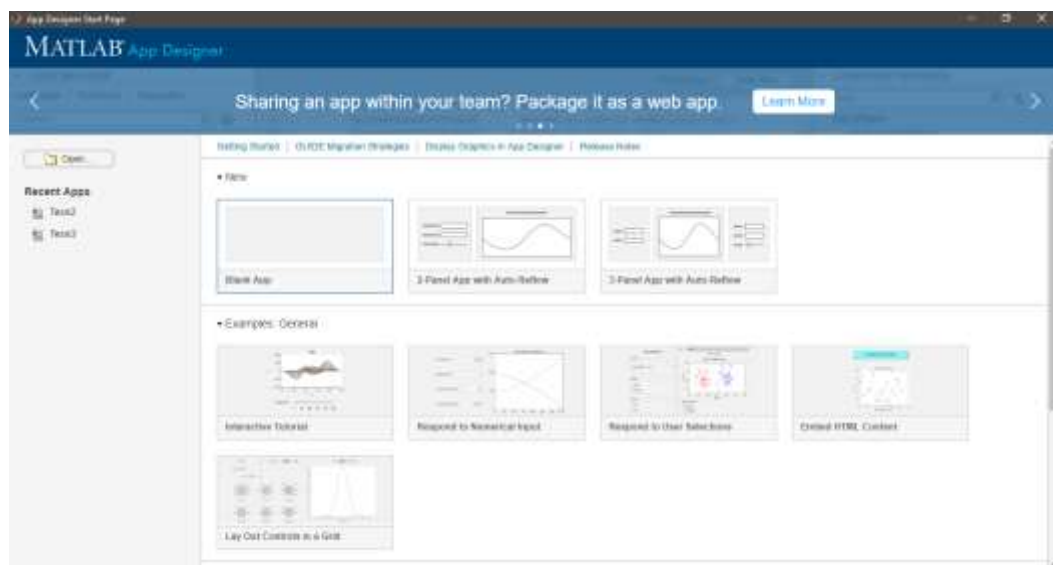
Nota: Por Miguel Escobar

Para acceder a MatLAB se debe ingresar *appdesigner* en el *Command Window*, de esta manera:

Figura 14 Comando para acceder a AppDesigner.

Nota: Por Miguel Escobar

Así MatLAB reconoce fácilmente la orden del usuario abriendo en una ventana independiente la plataforma de AppDesigner.

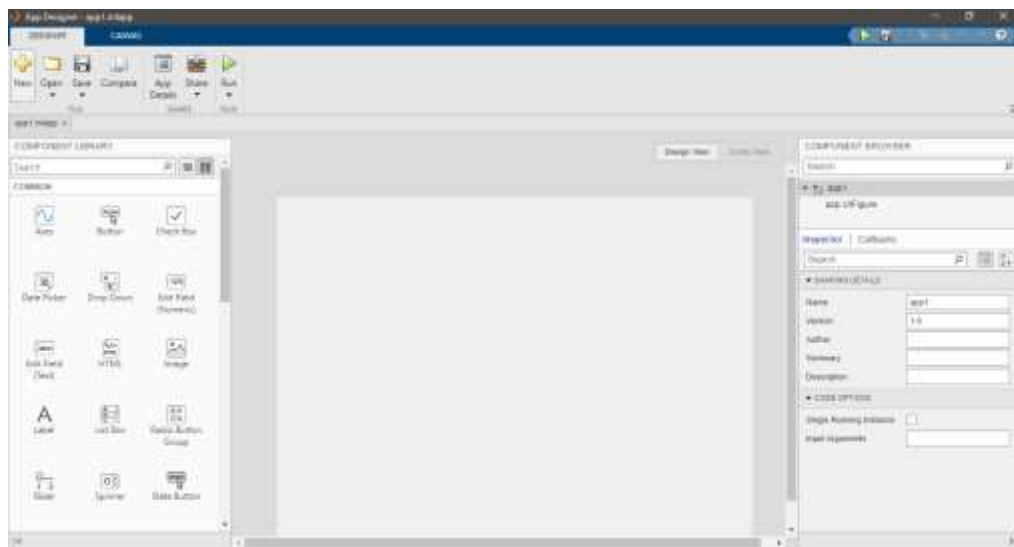
Figura 15.Interfaz de creación AppDesigner.

Nota: Por Miguel Escobar

Elementos de programación

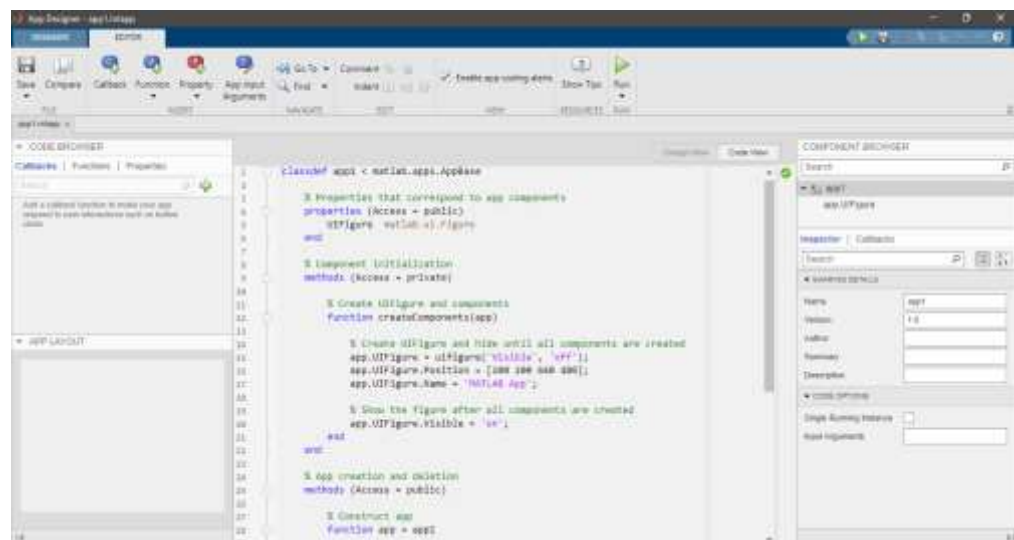
El lenguaje M que maneja AppDesigner es simple y muy amigable con el usuario, al acceder al *Blank App* se muestran las opciones de *Design View* y *Code View*, donde ambas comparten la pestaña de *Component Browser* donde se expone el nombre de las aplicaciones que son insertadas en la interfaz gráfica, aplicaciones que son posibles de insertar en la misma gracias a la pestaña *Component Library* que se despliega solo en la *Design View* del AppDesigner.

Figura 16. Appdesigner en Blank Mode.



Nota: Por Miguel Escobar

Figura 17. Code View AppDesigner

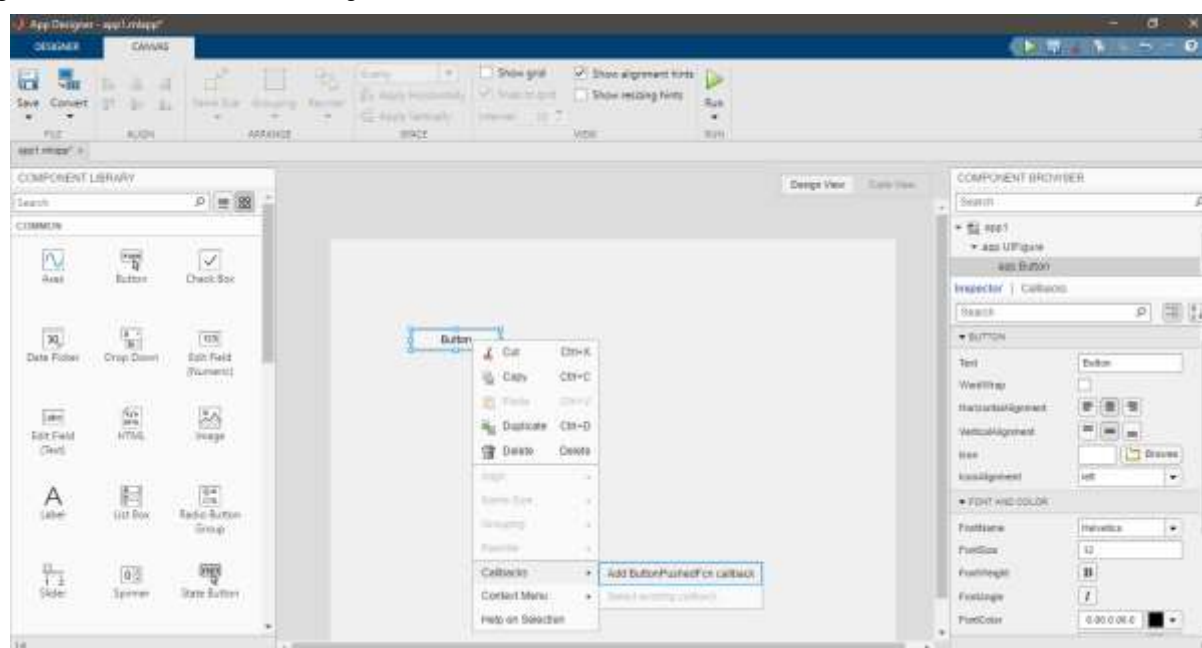


Nota: Por Miguel Escobar

La inserción de cualquier componente de la *Component Library* en la interfaz gráfica de la aplicación automáticamente crea un campo en la *Component Browser* nombrándola app."Elemento", de esta manera es como más adelante son recibidos los datos que se insertan en este elemento o los que se pretenden proyectar en el mismo.

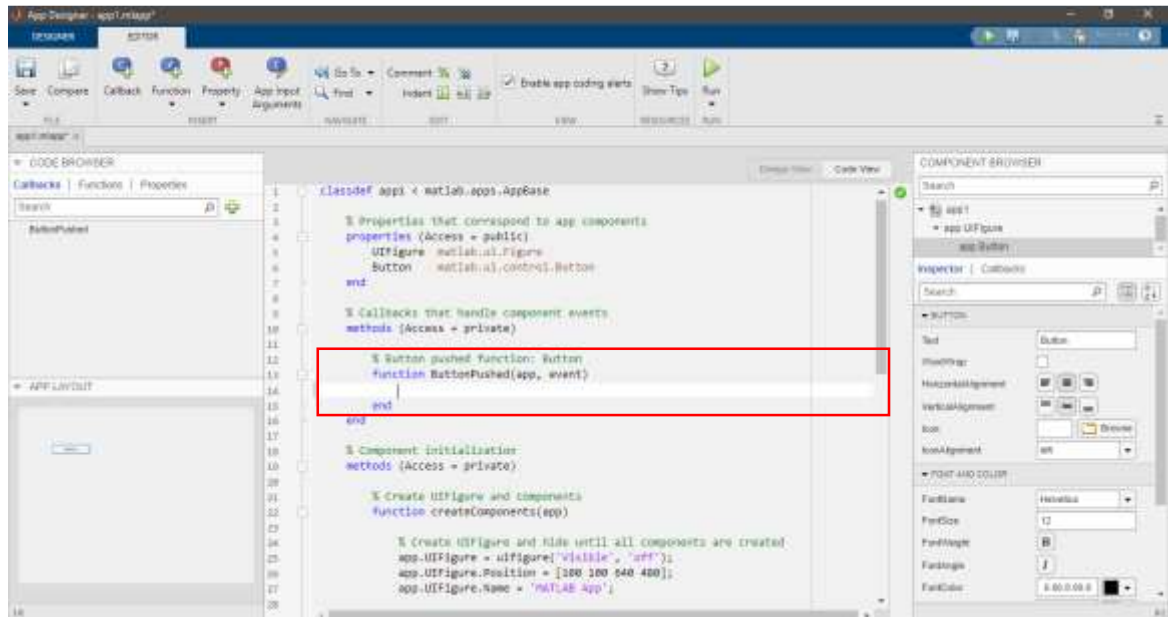
Los botones son los que tradicionalmente se usan para que el usuario de la orden de iniciar los cálculos que fueron programados, por lo tanto, son ellos por los que se inicia la programación creando en el sistema la orden del *callback*.

Figura 18. Orden Callback en Design View



Nota: Por Miguel Escobar

A la orden de *Add ButtonPushedFcn callback* por el usuario, el sistema automáticamente crea en el *Code View* una pequeña sección en blanco entre el código fuente del sistema en para la iniciación de la programación.

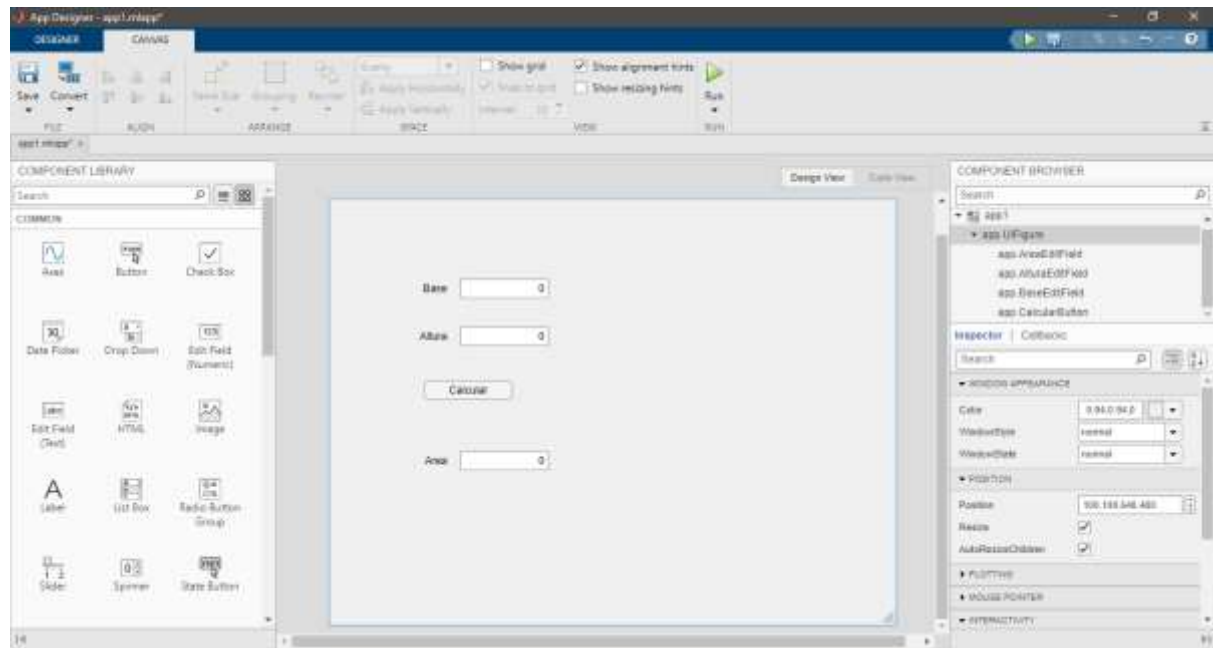
Figura 19. Orden Callback en Code View.

Nota: Por Miguel Escobar

La programación se compone de la sección de definición de variables, operaciones a realizar y publicación de resultados. Cada proceso de programación debe ser enumerada para la comprensión del orden por parte del sistema, asimismo, como cada variable que debe ser insertada por el usuario debe corresponder a una app de la interfaz e igualmente cada variable a publicar como resultado. A continuación, un ejemplo de una operación básica (área de cuadrado o rectángulo) para describir el funcionamiento y programación en el AppDesigner:

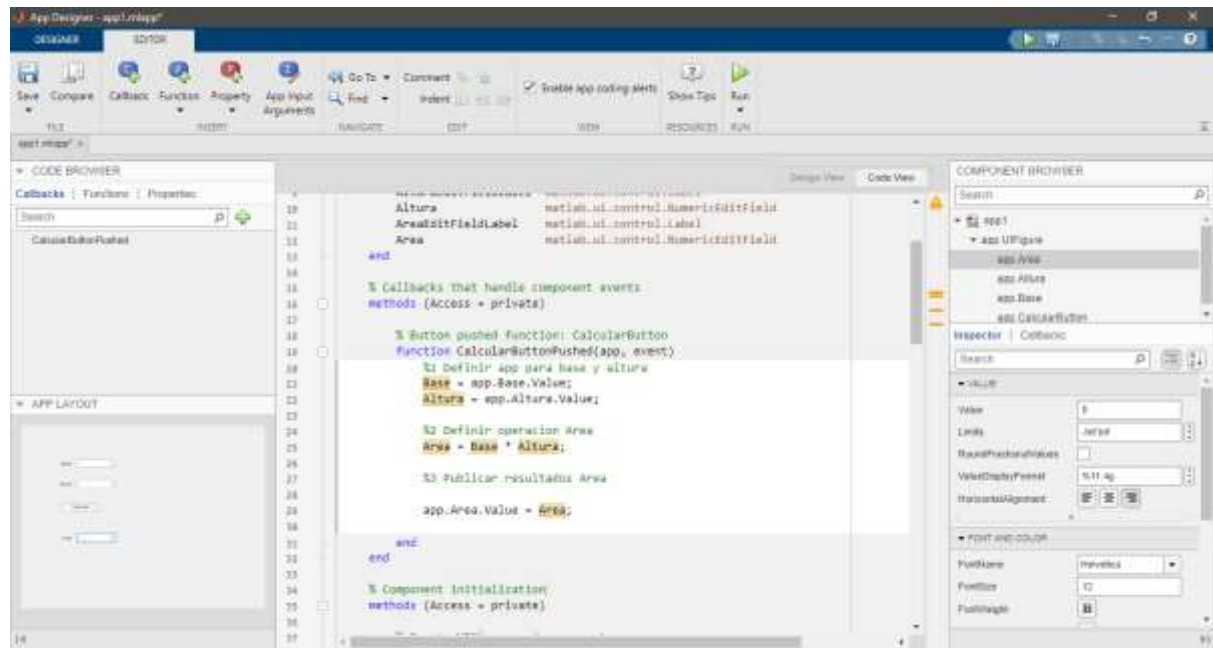
1. Se deben insertar los componentes necesarios y nombrarlos, en este caso dos deben ser insertados por el usuario, uno para la publicación del resultado y el botón para calcular.

Figura 20. Ejemplo calculo área en Design view.



Nota: Por Miguel Escobar

2. Se definen los pasos a leer por el sistema de la forma %1. Paso, %2. Paso, etc.
3. Se definen las variables a insertar nombrando las variables e igualarlas a las aplicaciones de inserción de texto, de la forma variable = app. variable. Value;
4. Las operaciones deben ser conformadas al pie de la letra de los nombres definidos en la lectura del sistema para su correcta interpretación. Resultado = variable^2;
5. La variable a publicar debe llevar la forma contraria a la definición del segundo paso, app.Resultado.Value = Resultado;

Figura 21. Code View del programa área.

Nota: Por Miguel Escobar

Etaapa Dos

La metodología Push-over es la obtención de la curva que determina la capacidad de la estructura hasta el momento de su rotura por parte de sus elementos estructurales. Para el desarrollo de este proyecto de investigación se planteó la construcción de las gráficas momento-curvatura y cortante basal-desplazamiento, de la siguiente manera:

Cálculo del momento curvatura

Para la obtención de la gráfica del momento curvatura el procedimiento fue el siguiente:

- Inercia del elemento en x

$$Inercia = \left(\frac{Base * Altura^3}{12} \right)$$

Ecuación 6 Inercia

- Módulo de elasticidad del elemento

$$Ec1 = fct/Ec$$

Ecuación 7 Modulo de elasticidad

- Momento fisurante

$$\text{Momento fisurante } (M1) = \frac{fct * Inercia}{Altura/2}$$

Ecuación 8. Momento fisurante.

- Φ fisurante

$$\Phi \text{ fisurante } (\Phi1) = \frac{Ec1}{\left(\frac{Altura}{2}\right)}$$

Ecuación 9. Φ fisurante

- n

$$n = \frac{Es}{Ec}$$

Ecuación 10. Valor n.

- Área del elemento

$$\text{Area} = \#varillas * \pi * \left(\#varilla * \frac{0.0254}{16}\right)^2$$

Ecuación 11. Area del elemento.

- Altura al centro del acero de refuerzo

$$d = \text{Altura} - \text{recubrimiento}$$

Ecuación 12. Altura al centro del acero de refuerzo

- Cuantía

$$p = \text{Area} / (\text{Base} * d)$$

Ecuación 13 Cuantía.

- X

$$x = d * n * p * (-1 + \sqrt{1 + 2/(n * cuantia)})$$

Ecuación 14. Valor x.

- G_s

$$G_s = \frac{2 * M_1}{Base * x * \left(d - \frac{x}{3}\right)}$$

Ecuación 15. G_s

- Módulo de elasticidad en límite de servicio

$$Ec_2 = \frac{G_s}{Ec}$$

Ecuación 16. Módulo de elasticidad en límite de servicio.

- Φ en límite de servicio

$$\Phi_2 = \frac{Ec_2}{x}$$

Ecuación 17. • Φ límite de servicio.

- G_{s2}

$$G_{s2} = \frac{(2 * Area * f_y)}{(Base * x)}$$

Ecuación 18. G_s límite de servicio

- Momento 1 en prevención de colapso

$$M_{3.1} = \frac{1}{2} * G_{s2} * Base * x * \left(d - \frac{x}{3}\right)$$

Ecuación 19. Momento uno prevención colapso.

- Módulo de elasticidad límite operativo

$$Es_1 = \frac{f_y}{Es}$$

Ecuación 20. Módulo de elasticidad límite operativo

- Φ_1 en prevención de colapso

$$\Phi_{3.1} = \frac{Es_1}{d - x}$$

Ecuación 21. • Φ uno en prevención colapso.

- x_2

$$x_2 = \frac{Area * f_y}{0.8 * f_c * Base}$$

Ecuación 22. Valor x_2 .

- Momento 2 en límite de colapso

$$M_{3.2} = x_2 * 0.8 * f_c * Base * \left(d - \left(0.8 * \frac{x_2}{2} \right) \right)$$

Ecuación 23. Momento dos en límite colapso.

- Φ_2 en límite de colapso

$$\Phi_{3.2} = \frac{Es_1}{d - x_2}$$

Ecuación 24. • Φ dos en límite colapso.

- Momento de límite de colapso, es la elección entre el valor máximo de $M_{3.1}$ y $M_{3.2}$.
- Momento de fisura o colapso es entonces el valor de Momento de límite de colapso (M_3)
- Φ de colapso

$$\Phi_4 = \frac{es_u}{d - x_2}$$

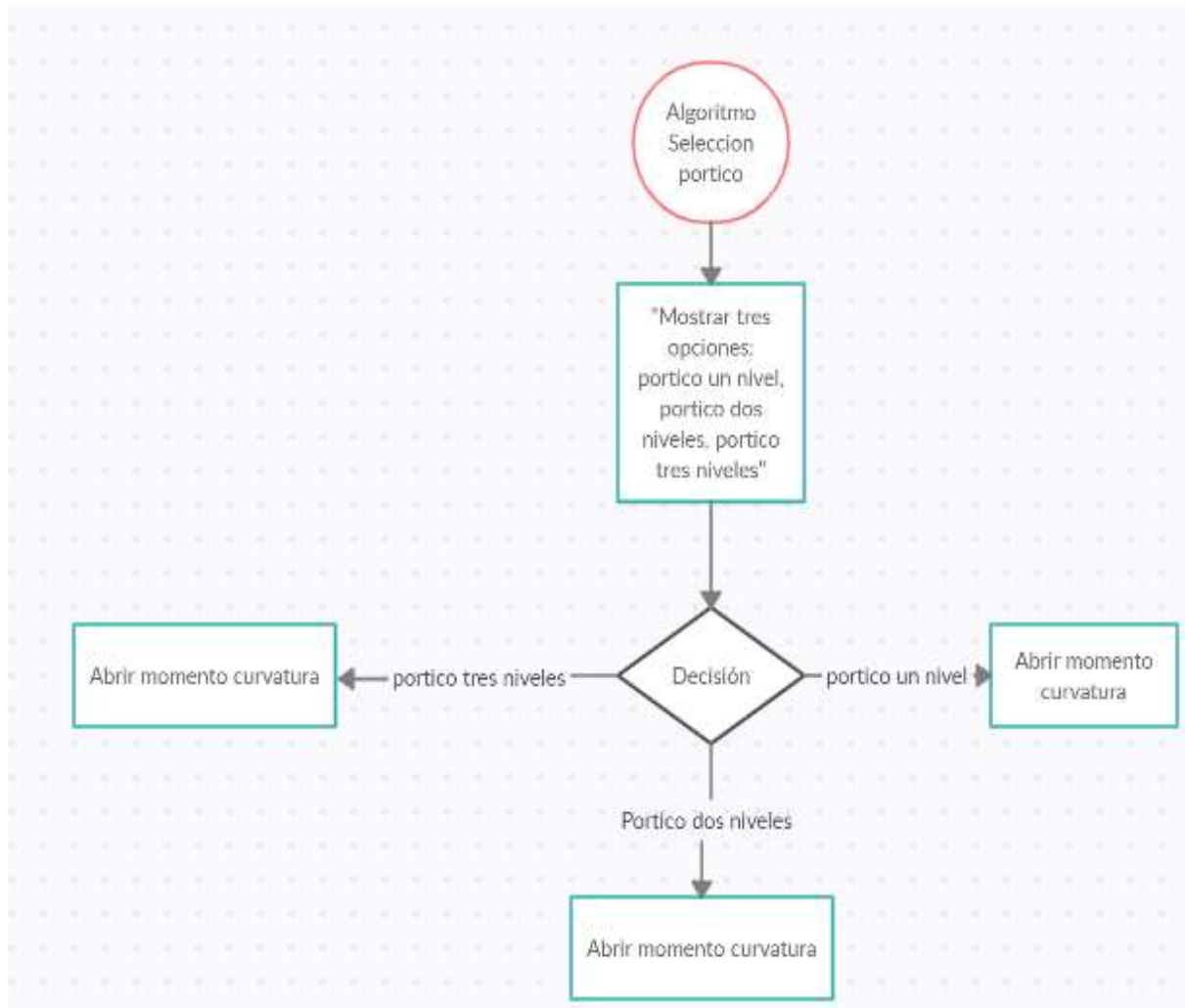
Ecuación 25. • Φ de colapso

Etapas Tres

Para la etapa tres se construyeron los algoritmos en el software libre Creately que servirían de base para la construcción del software educativo más adelante en el software de MatLAB.

Selección De Pórtico

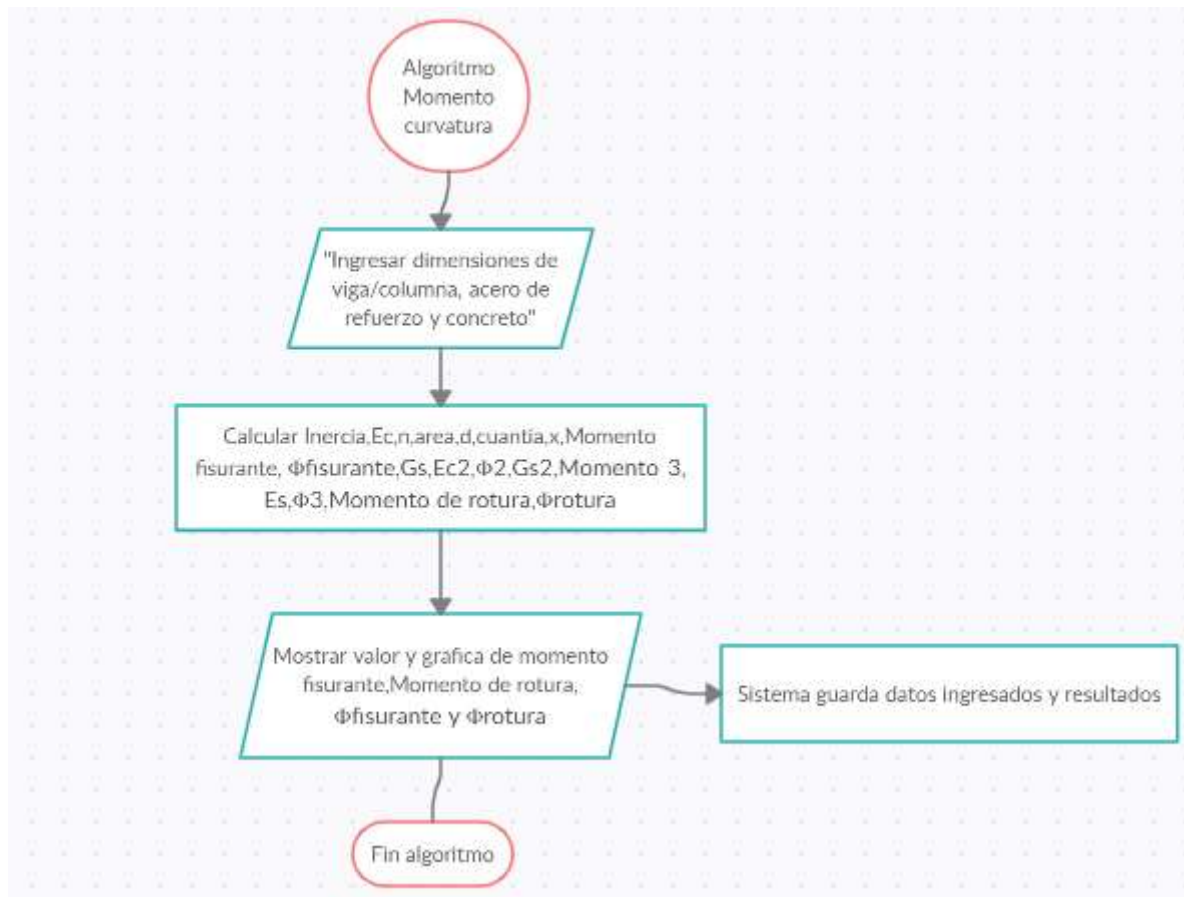
Este algoritmo describe la lógica que debe leer el software ante la decisión del usuario, donde se plantean tres redirecciones posibles las cuales son los tres tipos de pórticos que se le permiten analizar al usuario en el programa, siendo estos: pórtico simple en 2d de un nivel, pórtico simple en 2d de dos niveles y pórtico simple en 2d de tres niveles.

Figura 22. Algoritmo selección pórtico.

Nota: Por Miguel Escobar usando la herramienta creately.

Cálculo De Momento Curvatura

Después de ser seleccionado el tipo de pórtico el usuario es redirigido al cálculo del momento curvatura de las vigas y columnas del pórtico. Aquí el usuario inserta los datos que se le enuncian, Base, Altura y longitud de la viga y la columna, Numero de varillas y numero de la varilla de acero de refuerzo, F_{ct} , F_y , E_s , E_c , e_{su} , F_c y el recubrimiento. Con estos datos el usuario obtiene el cálculo de los momentos y fi fisurante y de rotura del elemento analizado y su correspondiente grafica de momento-curvatura.

Figura 23. Algoritmo Momento Curvatura.

Nota: Por Miguel Escobar usando la herramienta creately.

Debido a las limitaciones que posee el AppDesigner con respecto a la creación de una ventana ágil y fluida que permitiera los cálculos independientes de cada pórtico, se hizo necesario que en este algoritmo el sistema guardara en la memoria los datos insertados por el usuario para el desarrollo del tercer paso del método Push-over, el cálculo y grafico del cortante basal y el desplazamiento ocurrido por la fuerza horizontal.

Calculo Grafico Cortante Basal-Desplazamiento

A continuación, el usuario ingresa los datos que le suministra su espectro sísmico, esta información es solicitada para reconocer el periodo de aproximación y la pseudoaceleracion, datos que permiten el cálculo de la rigidez de los elementos, el cortante basal y las fuerzas horizontales.

Figura 24. algoritmo cortante basal-desplazamiento

Nota: Por Miguel Escobar usando la herramienta createely.

Problema Base

Para la realización del programa se planteó la posibilidad de solución de pórticos simples en 2d de hasta tres niveles con la misma configuración de columnas y vigas para todos los pisos que se inserta en el desarrollo del segundo algoritmo previamente expuesto.

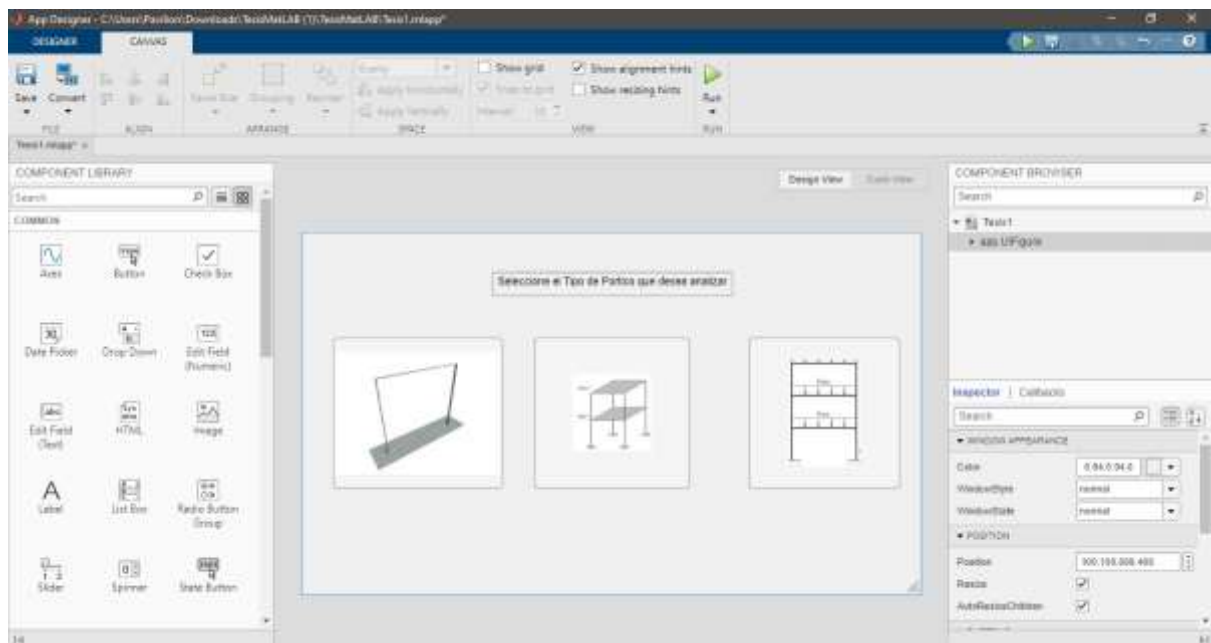
Resultados

Selección De Pórtico

Para la selección del pórtico se creó una ventana independiente para la selección del pórtico a gusto del usuario, con tres botones a los que se les programo una intercomunicación de ventanas para acceder al siguiente paso, cálculo de momento-curvatura, la interfaz se diseñó para ser amigable y clara para el usuario.

Interfaz

Figura 25. Interfaz selección de pórtico.



Nota: Por Miguel Escobar.

Calculo Momento-Curvatura

Para el cálculo del momento-curvatura se creó una interfaz gráfica de una ventana con la opción *scrollable* del menú *interactive* activada para que al momento de correr la aplicación debido a su magnitud pueda ser desplazado por el usuario para la obtención de gráficas y resultados.

Interfaz

Figura 26. Interfaz calculo momento-curvatura.

Nota: Por Miguel Escobar.

Calculo Cortante Basal-Desplazamiento

Para el cálculo de la gráfica del cortante basal-desplazamiento nuevamente se construyó una interfaz con la opción *scrollable* activada en los casos de pórticos de dos niveles y tres niveles debido a su extensión. Para todos los casos se muestran los datos que vienen de la fase de momento-curvatura en la parte superior de la pantalla de manera que se agilice el proceso de cálculo y se evite que por causa de las limitaciones del software el usuario no tenga que replicar datos.

Interfaz Primer Nivel

Figura 27. Interfaz cortante basal-desplazamiento pórtico un nivel.

Nota: Por Miguel Escobar.

Interfaz segundo nivel

Figura 28. Calculo cortante basal-desplazamiento dos niveles.

Nota: Por Miguel Escobar.

Análisis De Resultados

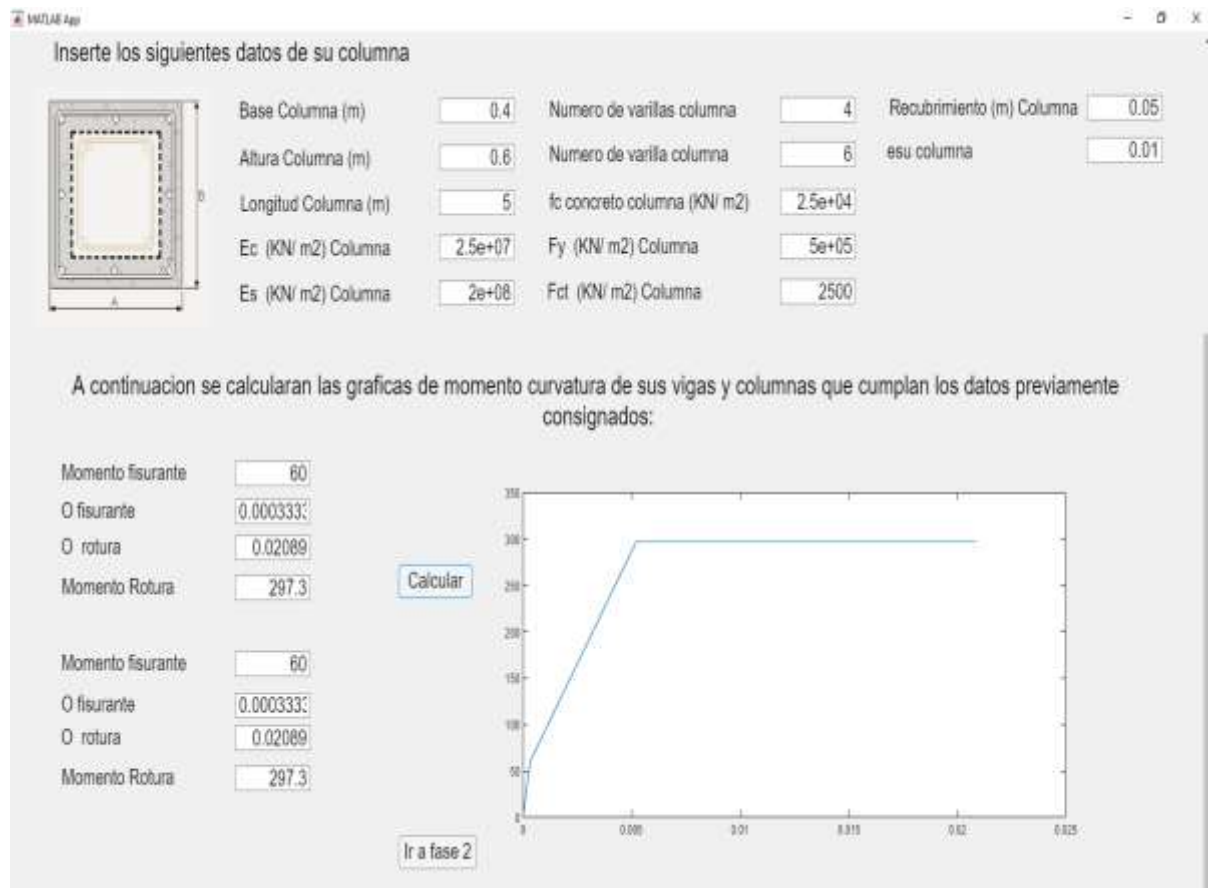
Para comprobar que los resultados que presenta el software didáctico se plantearon datos iniciales de los cuales se reconocen los resultados gracias a su elaboración en el software Microsoft Office Excel y en colaboración con la teoría impartida en Estructuras III por parte del Dr. Antoni Cladera, docente de la Universitat de les Illes Balears (UIB).

Cálculo De Momento Curvatura

Los datos iniciales correspondientes a las dimensiones y caracterización de las vigas y columnas fue la siguiente:

- Base de 0.4m
- Altura de 0.6m
- Longitud de 5m
- Numero de varillas 4
- Varilla #6
- Recubrimiento de 0,05m
- ϵ_{su} de 0,01
- F_{ct} 2500 Kn/m²
- F_c concreto 25.000 Kn/m²
- F_y 500.000 Kn/m²
- E_s 200'000.000 Kn/m²
- E_c 25'000.000 Kn/m²

Debido a que los valores que se insertaron eran iguales para viga y columna los resultados obtenidos fueron los siguientes:

Figura 30. Comprobación resultados momento-curvatura

Nota: Por Miguel Escobar.

Donde evidenciamos que los resultados obtenidos por el docente y los suministrados por AppDesigner concuerdan por lo que la obtención de la gráfica momento-curvatura de los elementos estructurales, por lo tanto, no se necesitaron correcciones y ofrece resultados ajustados a la teoría y a la realidad.

Calculo Cortante Basal-Desplazamiento

Para el chequeo del cálculo y grafica del cortante basal y el desplazamiento de todos los tipos de pórtico disponibles por el software se predeterminaron los siguientes valores a insertar por el usuario, la teoría manejada para el chequeo de los resultados fue la recibida personalmente a lo largo de la asignatura de concreto reforzado y patología de la construcción.

- Aa de 0.35
- Av de 0.3
- Fa de 1,15

- Fv de 1,8
- I de 1
- γ concreto de 24 Kn/m³

Resultados Pórtico De Un Nivel

Tabla 1. Datos y Cálculos pseudoaceleración.

Pseudo aceleración (Espectro)	
Aa	0,35
Av	0,3
Fa	1,15
Fv	1,8
I	1
Sa	1,00625 g
Ct	0,047
a	0,9
Altura total	5
Periodo aproximado (Ta)	0,20 seg
K	1

Nota: Por Miguel Escobar.

Tabla 2. Datos y cálculos cortante basal y fuerza horizontal.

Cortante basal			Vb	57,96	Kn
Piso	Altura	Peso	W*H ^k	Cvi	Fi (x)
1	5,00 m	57,60 Kn	288,00 Kn	1	57,96 Kn
Sumatoria	-	57,60 Kn	288,00 Kn	-	-

Nota: Por Miguel Escobar.

Tabla 3. *Análisis a columnas primer piso*

ANÁLISIS A COLUMNAS			
Modulo elasticidad	E	25000000,00 MPa	25000000,00 KPa
Inercia X	Ixx		0,00720 m4
Inercia Y	Iyy		0,00320 m4
Altura total piso	H		5,00 m
Rigidez columna en X	KX col		7680,00 Kn/m
Rigidez columna en Y	KY col		17280,00 Kn/m
Rigidez estructura en X	KX est		15360,00 Kn/m
Desplazamiento en X	Des X		0,00377 m
Deriva en X	Der X		0,075 %

Nota: Por Miguel Escobar.

Figura 31. *Cálculos primer nivel cortante basal-desplazamiento.*

Base Columna: 0.4 Altura Columna: 0.6 Longitud Columna: 5 $\Phi 1$: 0.0003333 F_c : 2.5e+04

Base Viga: 0.4 Altura viga: 0.6 Longitud Viga: 5 $\Phi 2$: 0.02089 E_c : 2.5e+07

A_a : 0.35 S_a : 1.006 Rigidez en x (Kn/m): 1.536e+04

A_v : 0.3 Periodo aproximado: 0.2001 Rigidez en y (Kn/m): 3.456e+04

F_a : 1.15 K: 1 Desplazamiento en X: 0.003773

F_v : 1.8 Peso de la estructura (Kn): 57.6 Desplazamiento en Y: 0.001677

I : 1 Cortante basal (Kn): 57.96 Desplazamiento Eductil: 0.003773

$\gamma_{concreto}$ (KN/m3): 24 F_x (Kn): 57.96 Desplazamiento Erotura: 0.1065

F_y (Kn): 57.96

Nota: Por Miguel Escobar.

Resultados Pórtico Dos Niveles**Tabla 4.** *Pseudoaceleración.*

Pseudo aceleración (Espectro)	
Aa	0,35
Av	0,3
Fa	1,15
Fv	1,8
I	1
Sa	1,00625 g
Ct	0,047
a	0,9
Altura total	10,00
Periodo aproximado (Ta)	0,37 seg
K	1

Nota: Por Miguel Escobar.

Tabla 5. *Cortante basal y fuerza horizontal.*

Cortante basal			Vb	144,90 Kn	
Piso	Altura	Peso	W*H^k	Cvi	Fi (x)
1	5,00 m	86,40 Kn	432,00 Kn	0,6	86,94 Kn
2	5,00 m	57,60 Kn	288,00 Kn	0,4	57,96 Kn
Sumatoria	-	144,00 Kn	720,00 Kn	1,00	-
Piso	Altura	Peso	W*H^k	Cvi	Fi (y)
1	5,00 m	86,40 Kn	432,00 Kn	0,6	86,94 Kn
2	5,00 m	57,60 Kn	288,00 Kn	0,4	57,96 Kn
Sumatoria	-	144,00 Kn	720,00 Kn	1	-

Nota: Por Miguel Escobar.

Tabla 6. *Analisis a columnas piso 1.*

ANÁLISIS A COLUMNAS PISO 1			
Esfuerzo a compresión	F'c	21 MPa	
Modulo elasticidad	E	25000000,00 MPa	25000000,00 KPa
Inercia X	Ixx	0,00720 m4	
Inercia Y	Iyy	0,00320 m4	
Altura total piso	H	5,00 m	
Rigidez columna en X	KX col	7680,00 Kn/m	
Rigidez columna en Y	KY col	17280,00 Kn/m	
Rigidez estructura en X	KX est	15360,00 Kn/m	
Rigidez estructura en y	KY est	34560,00 Kn/m	

Nota: Por Miguel Escobar.

Tabla 7. *Análisis a columnas piso 2*

ANÁLISIS A COLUMNAS PISO 2			
Esfuerzo a compresión	F'c	21 MPa	
Modulo elasticidad	E	25000000,00 MPa	25000000,00 KPa
Inercia X	Ixx	0,00720 m4	
Inercia Y	Iyy	0,00320 m4	
Altura total piso	H	5,00 m	
Rigidez columna en X	KX col	7680,00 Kn/m	
Rigidez columna en Y	KY col	17280,00 Kn/m	
Rigidez estructura en X	KX est	15360,00 Kn/m	
Rigidez estructura en y	KY est	34560,00 Kn/m	

Nota: Por Miguel Escobar.

Figura 32 *Calculo y resultados cortantes basal-desplazamiento.*

Datos iniciales

Base Columna Altura Columna Longitud Columna $\phi 1$ F_c

Base Viga Altura viga Longitud Viga $\phi 2$ E_c

Inserte los siguientes datos (procedentes de su espectro sismico de diseño)

A_a A_v F_a F_v I

$\gamma_{concreto}$ (KN/m³)

Calculos primer nivel

Peso del primer nivel (Kn)

Cortante basal (Kn)

F_x (Kn)

Desplazamiento en X

Desplazamiento θ ductil

Desplazamiento θ rotura

Calculos segundo nivel

Peso del segundo nivel (Kn)

Cortante basal (Kn)

F_x (Kn)

Desplazamiento en X

Desplazamiento θ ductil

Desplazamiento θ rotura

Calculos estructurales

S_a Periodo aproximado K Cortante basal (Kn)

Peso total de la estructura (Kn)

Nota: Por Miguel Escobar.

Resultados Pórtico Tres Niveles

Tabla 8. *Cálculos pseudoaceleracion.*

Pseudo aceleración (Espectro)	
A_a	0,35
A_v	0,3
F_a	1,15
F_v	1,8
I	1
S_a	1,00625 g
C_t	0,047
a	0,9
Altura total	15,00
Periodo aproximado (T_a)	0,54 seg
K	1,018874735

Nota: Por Miguel Escobar.

Tabla 9. *Análisis a columnas del piso 1.*

ANÁLISIS A COLUMNAS PISO 1			
Esfuerzo a compresión	F'c	21 MPa	
Modulo elasticidad	E	25000000,00 MPa	25000000,00 KPa
Inercia X	Ixx	0,00720 m ⁴	
Inercia Y	Iyy	0,00320 m ⁴	
Altura total piso	H	5,00 m	
Rigidez columna en X	KX col	7680,00 Kn/m	
Rigidez columna en Y	KY col	17280,00 Kn/m	
Rigidez estructura en X	KX est	15360,00 Kn/m	
Rigidez estructura en y	KY est	34560,00 Kn/m	

Nota: Por Miguel Escobar.

Tabla 10. *Análisis a columnas piso 2.*

ANÁLISIS A COLUMNAS PISO 2			
Esfuerzo a compresión	F'c	21 MPa	
Modulo elasticidad	E	25000000,00 MPa	25000000,00 KPa
Inercia X	Ixx	0,00720 m ⁴	
Inercia Y	Iyy	0,00320 m ⁴	
Altura total piso	H	5,00 m	
Rigidez columna en X	KX col	7680,00 Kn/m	
Rigidez columna en Y	KY col	17280,00 Kn/m	
Rigidez estructura en X	KX est	15360,00 Kn/m	
Rigidez estructura en y	KY est	34560,00 Kn/m	

Nota: Por Miguel Escobar.

Tabla 11. *Análisis a columnas piso 3*

ANÁLISIS A COLUMNAS PISO 3			
Esfuerzo a compresión	F'c	21 MPa	
Modulo elasticidad	E	25000000,00 MPa	25000000,00 KPa
Inercia X	Ixx	0,00720 m4	
Inercia Y	Iyy	0,00320 m4	
Altura total piso	H	5,00 m	
Rigidez columna en X	KX col	7680,00 Kn/m	
Rigidez columna en Y	KY col	17280,00 Kn/m	
Rigidez estructura en X	KX est	15360,00 Kn/m	
Rigidez estructura en y	KY est	34560,00 Kn/m	

Nota: Por Miguel Escobar.

Tabla 12. *Cortante basal y fuerzas horizontales.*

Cortante basal			Vb	231,84 Kn	
Piso	Altura	Peso	W*H^k	Cvi	Fi (x)
1	5,00 m	86,40 Kn	445,32 Kn	0,375	86,94 Kn
2	5,00 m	86,40 Kn	445,32 Kn	0,375	86,94 Kn
3	5,00 m	57,60 Kn	296,88 Kn	0,25	57,96 Kn
Sumatoria	-	230,40 Kn	1187,53 Kn	1	-
Piso	Altura	Peso	W*H^k	Cvi	Fi (y)
1	5,00 m	86,40 Kn	445,32 Kn	0,375	86,94 Kn
2	5,00 m	86,40 Kn	445,32 Kn	0,375	86,94 Kn
3	5,00 m	57,60 Kn	296,88 Kn	0,25	57,96 Kn
Sumatoria	-	230,40 Kn	1187,53 Kn	1	-

Nota: Por Miguel Escobar.

Figura 33. *Calculo cortante basal y desplazamiento.*

Datos iniciales

Base Columna Altura Columna Longitud Columna $\phi 1$ F_c

Base Viga Altura viga Longitud Viga $\phi 2$ E_c

Inserte los siguientes datos (procedentes de su espectro sismico de diseño)

Aa Av Fa Fv I

$\gamma_{concreto}$ (KN/m³)

Calculos primer nivel

Peso del primer nivel (Kn)

Cortante basal (Kn)

Fx (Kn)

Desplazamiento en X

Desplazamiento θ_{ductil}

Desplazamiento θ_{rotura}

Calculos segundo nivel

Peso del segundo nivel (Kn)

Cortante basal (Kn)

Fx (Kn)

Desplazamiento en X

Desplazamiento θ_{ductil}

Desplazamiento θ_{rotura}

Nota: Por Miguel Escobar.

Figura 34. *Calculo cortante basal y desplazamiento.*

Calculos tercer nivel

Peso del tercer nivel (Kn)

Cortante basal (Kn)

Fx (Kn)

Desplazamiento en X

Desplazamiento θ_{ductil}

Desplazamiento θ_{rotura}

Calculos estructurales

Sa Periodo aproximado K Cortante basal (Kn)

Peso total de la estructura (Kn)

Nota: Por Miguel Escobar.

Se observa que los resultados obtenidos concuerdan entre el Excel y los que expone el programa. Determinando así que el software educativo está ofreciéndole al usuario resultados

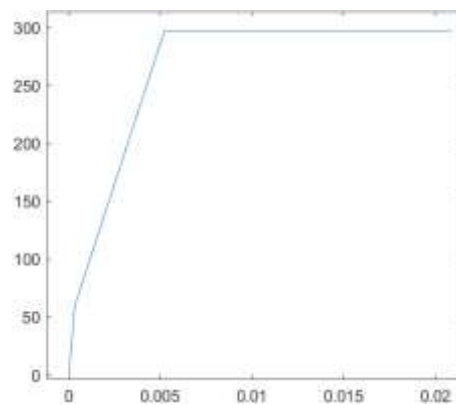
que describen correctamente el comportamiento de un pórtico simple de hasta tres niveles a través del método Push-over.

Graficas Obtenidas

Momento Curvatura

En la figura 35 podemos observar el momento curvatura de la estructura analizada, con los datos de la curvatura obtenidos en metros en el eje de las abscisas y el momento presente en el eje de las ordenadas.

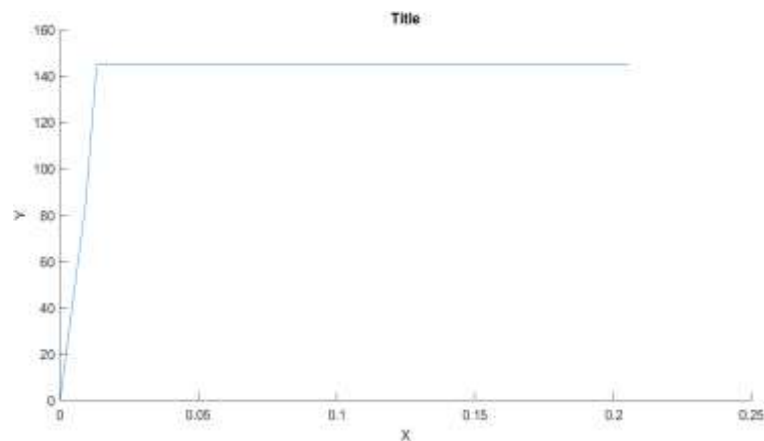
Figura 35. Grafica momento curvatura.



Nota: Por Miguel Escobar.

Cortante Basal Desplazamiento Pórtico Dos Niveles

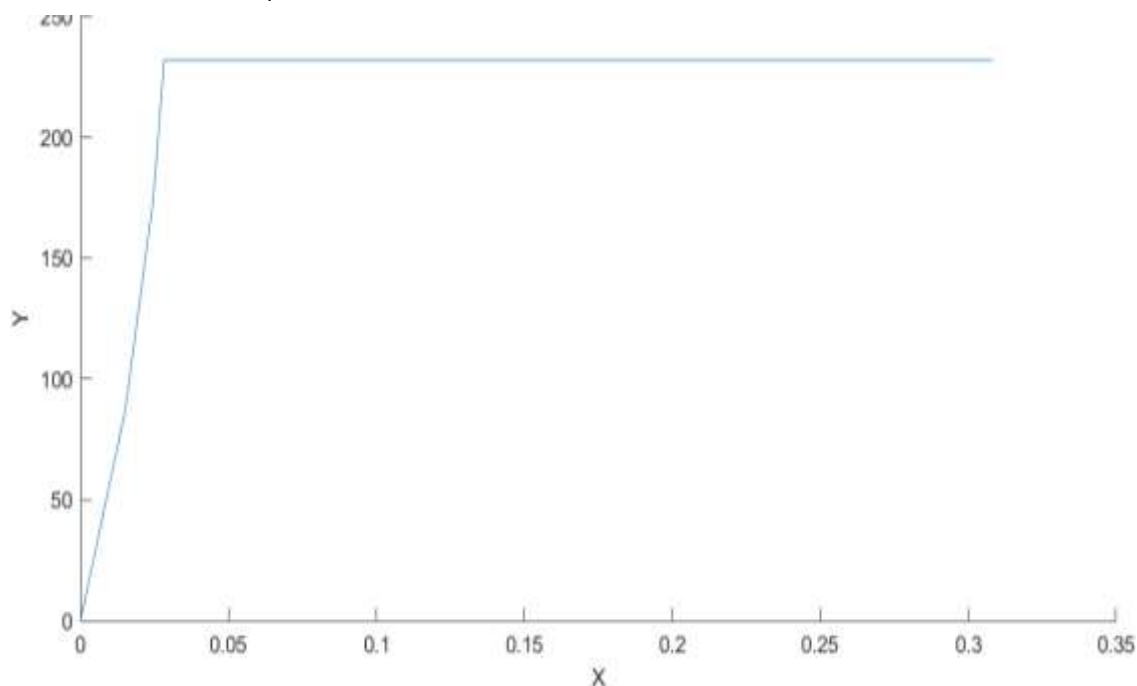
Figura 36 Cortante-Basal desplazamiento dos niveles.



Nota: Por Miguel Escobar.

Cortante Basal-Desplazamiento Pórtico Tres Niveles

Figura 37 Cortante basal-desplazamiento tres niveles.



Nota: Por Miguel Escobar.

Se puede observar a través de las gráficas el comportamiento de los pórticos simples bajo el análisis no lineal que ofrece la metodología Push-over. Que tanto en el pórtico de dos niveles como de tres niveles que cumplen con los máximos permitidos en desplazamientos horizontales debido a que sus derivas (calculadas en Excel) no superan el 1% que habilita la NSR-10%:

Tabla 13. Derivas del análisis de dos pisos.

Der. Piso 1	0,1886719 %
Der. Piso 2	0,0754688 %

Nota: Por Miguel Escobar.

Tabla 14. Derivas del analisis de tres pisos.

Der. Piso 1	0,3018750 %
Der. Piso 2	0,1886719 %
Der. Piso 3	0,0754688 %

Nota: Por Miguel Escobar.

De esta forma se expone el funcionamiento claro, sencillo y eficaz que ofrece el software educativo, de manera que los involucrados en el proceso de enseñanza-aprendizaje se les sea sencillo obtener los datos resultados y su interpretación, a través de las gráficas, sea posible involucrarla en otros componentes del análisis estructural como lo son el cálculo de las derivas.

Conclusiones Y Trabajos Futuros

Conclusiones

Se logro a través del programa desarrollado a lo largo de este proyecto de investigación la puesta en digital de manera automatizada del método de análisis no lineal Push-over que permite la obtención de la curva capacidad que expone los puntos de desplazamiento dúctiles y de rotura en concordancia con el cortante basal que se presenta en la estructura a través de los parámetros que sufre de acuerdo con su espectro sísmico.

El producto obtenido gracias a la interfaz de programación AppDesigner se define como una UI amigable y sencilla para el público objetivo, los estudiantes, donde todos sus elementos permiten la lectura rápida y clara de que se pide y que se resuelve.

Trabajos Futuros

Este proyecto de investigación permite que más estudiantes lo utilicen como una base para la creación de programas que usen la plataforma de AppDesigner como Interfaz gráfica programable debido a que se expone de manera clara y concisa la forma en la que se programa y el cómo se utilizan todos sus apartados de un modo resumido y amigable para cualquier novato en el tema.

Así mismo, como se expone en el alcance, se puede usar este proyecto como punto base para partir en dos direcciones: la mejora del programa en si para el cálculo del método Push-over para estructuras más complejas como lo son pórticos simples en 3d de n niveles o con cargas establecidas de forma que su enfoque no sea educativo sino profesional. La segunda dirección puede ser tomar la experiencia adquirida en el desarrollo del proyecto para la creación de un programa que presente otra metodología estructural distinta al método Push-over ya sea con énfasis profesional o educativo.

Bibliografía

- Bohorquez, H. A. (2016). *El punto de desempeño en porticos resistentes segun la ATC-40*. Santiago de Cali: [Trabajo de aula, Universidad del Valle]. Obtenido de <https://es.slideshare.net/habohorquez/el-punto-de-desempeo-en-porticos-resistentes-a>
- Calcina Peña, R. M. (2017). *Evaluacion del desempeño sismico de un edificio de once pisos utilizando analisis estatico y dinamico no lineal*. Tacna: [Trabajo de grado, Universidad privada de Tacna]. Obtenido de http://repositorio.upt.edu.pe/bitstream/handle/UPT/499/Calcina_Pena_Ramon.pdf;jsessionid=0DB438D056FEDEDF0015BACD0F3F4705?sequence=1
- Comunidad para la Ingenieria Civil. (2015). *Pushover-Analisis no lineal estatico en edificios de concreto armado*. Culiacan: CIC.
- Diaz, R. B. (2016). Desempeño sismico de edificios: binomio capacidad-demanda. En R. B. Diaz, *Vulnerabilidad y riesgo sismico de edificios* (págs. 41-98). Medellin. Obtenido de <https://www.tdx.cat/bitstream/handle/10803/6230/04CAPITULO3.pdf?sequence=4&isAllowed=y>
- Duarte Bonilla, C. E., Martínez Chavarría, M. E., & Santamaría Díaz, J. J. (2017). *Analisis estatico no lineal (Pushover) del cuerpo central del edificio de la facultad de medicina de la Universidad de El Slavador*. Ciudad Universitaria: [Trabajo de licenciatura, Universidad de El Salvador]. Obtenido de <http://ri.ues.edu.sv/id/eprint/13405>
- Gutierrez, A. (2015). *Notas sobre el Metodo de Analisis No lineal*. ALACERO. Obtenido de https://www.alacero.org/sites/default/files/u16/ci_23_-_21_notas_sobre_el_metodo_de_analisis_no_lineal.pdf
- Jorquera, A. Z. (2017). *Modelado y analisis Pushover de la respuesta sismica de estructuras de acero*. Cartagena: [Trabajo de grado, Universidad Politecnica de Cartagena]. Obtenido de <https://repositorio.upct.es/xmlui/bitstream/handle/10317/6092/tfm-zab-mod.pdf?sequence=1&isAllowed=y>
- Moller, O., Rubinstein, M., & Ascheri, J. P. (2011). *Analisis del amortiguamiento proporcional a la rigidez tangente en sistemas dinamicos no lineales*. Rosario: [Articulo, Asociacion Argentina de Mecanica Computacional]. Obtenido de <https://cimec.org.ar/ojs/index.php/mc/article/view/3826>

- Pappalardo, M. G. (2017). *Escuela tradicional: características, ventajas y desventajas*. Ciudad de Mexico: Lifeder. Obtenido de <https://www.lifeder.com/escuela-tradicional/>
- Possos Pedroza, G. A. (2014). *Software educativo para el diseño de vigas isostaticas de concreto presforzado, usando programacion en Visual Basic*. Bogota: [Trabajo de grado, Universidad Santo Tomas]. Obtenido de <https://repository.usta.edu.co/bitstream/handle/11634/708/Software%20educativo%20para%20el%20diseno%20de%20vigas%20isostaticas%20de%20concreto%20presforzado,%20usando%20programacion%20en%20visual%20basic.pdf?sequence=1&isAllowed=y>
- Salcedo Barco , Y. H. (2017). *Desempeño sismico por el metodo Pushover de un edificio construido en la ciudad de Barranquilla bajo la norma NSR-10*. Barranquilla: [Trabajo de grado, Universidad de la Costa]. Obtenido de <https://repositorio.cuc.edu.co/bitstream/handle/11323/248/1032418668.pdf?sequence=1&isAllowed=y>
- Solana, G., Arias, L., & Gomez, D. (2012). *Ventajas del empleo del software educativos en el proceso de enseñanza-aprendizaje en la universidad*. Buenos Aires: [Revista Digital Buenos Aires]. Obtenido de <https://www.efdeportes.com/efd171/ventajas-del-empleo-del-software-educativo.htm>
- Universidad Autonoma de Mexico. (2018). *Propiedades de los materiales en secciones de concreto reforzado*. Mexico: Facultad de Ingenieria. Obtenido de <https://docplayer.es/88935697-Propiedades-de-los-materiales-en-secciones-de-concreto-reforzado-capitulo-2-propiedades-de-los-materiales-en-secciones-de-concreto-reforzado.html>
- Valencia Vasconez, J. D. (2012). *Analisis del metodo modal pushover y su incidencia en el calculo de porticos de hormigon armado en el Canton de Ambato*. Ambato: [Trabajo de grado, Universidad Tecnica de Ambato]. Obtenido de <https://www.udocz.com/co/read/19977/tesis---analisis-del-metodo--modal-push-over-y-su-incidencia-en-el-calculo-de-porticos-de-hormigon-armado-en-el-canton-de-ambato->

Anexos

A continuación, se presentan las líneas de código que hicieron posible el funcionamiento óptimo del software educativo:

Código De Programación Selección de pórtico

1. Si se selecciona el pórtico de un nivel:

```
% Button pushed function: Button
function ButtonPushed2(app, event)
    Tesis3
end
```

2. Si se selecciona el pórtico de dos niveles:

```
% Button pushed function: Button_2
function Button_2Pushed (app, event)
    Tesis2niveles
end
```

3. Si se selecciona el pórtico de tres niveles:

```
% Button pushed function: Button_3
function Button_3Pushed (app, event)
    Tesis3niveles
end
```

Código De Programación Calculo momento curvatura

```
% Button pushed function: Calcular
function CalcularButtonPushed(app, event)
    %1. Asignar Variables
    Bviga = app.BaseVigam.Value;
    Hviga = app.AlturaViga.Value;
    Lviga = app.LongitudViga.Value;
    NoVarillas = app.NumerodevarillasViga.Value;
    fcviga = app.fcconcretovigaKNm2.Value;
    Ecviga = app.EcKNm2Viga.Value;
    fyviga = app.FyKNm2Viga.Value;
    fctviga = app.FctKNm2Viga.Value;
    Esviga = app.EsKNm2Viga.Value;
    recubrimientoviga = app.Recubrimientomviga.Value;
    numerovarilla = app.Numerodevarillaviga.Value;
```

```

Bcolumna = app.BaseColumna.Value;
Hcolumna = app.AlturaColumna.Value;
Lcolumna = app.LongitudColumna.Value;
NoVarillasc = app.Numerodevarillascolumna.Value;
fccolumna = app.fcconcretocolumnaKNm2.Value;
Eccolumna = app.EcKNm2Columna.Value;
fycolumna = app.FyKNm2Columna.Value;
fctcolumna = app.FctKNm2Columna.Value;
Escolumna = app.EsKNm2Columna.Value;
recubrimientocolumna = app.RecubrimientomColumna.Value;
numerovarillac = app.Numerodevarillacolumna.Value;
esuviga = app.esuviga.Value;
esucolumna = app.esucolumna.Value;

```

%2 Definir valores para Iviga y Ec1

```

Iviga = ((Bviga*(Hviga^3))/12);
Ec1 = fctviga/Ecviga;

```

%3 mostrar calculo de Momento curvatura viga fisura

```

app.Momentofisurante.Value = (fctviga * Iviga)/(Hviga/2);
app.Ofisurante.Value = Ec1/(Hviga/2);

```

%4 Definir valores de , Ec2, x, M1, M2, M3, M31,M32, M4, Fi1, Fi2, Fi31, Fi32, Fi4, pviga, areaviga, dviga, Gsviga, nviga,Gs2viga,x2,Esviga1

```

nviga = Esviga / Ecviga;
areaviga = NoVarillas * pi * (numerovarilla * 0.0254/16)^2;
dviga = Hviga - recubrimientoviga;
pviga = areaviga/(Bviga * dviga);
x = dviga * nviga * pviga * (-1+ sqrt((1+(2/(nviga*pviga)))));
M1 = app.Momentofisurante.Value;
Fi1 = app.Ofisurante.Value;
Gsviga = (2*M1)/(Bviga*x*(dviga-(x/3)));
Ec2 = Gsviga/Ecviga;
Fi2 = Ec2/x;
Gs2viga = (2*areaviga*fyviga)/(Bviga*x);
M31 = (1/2)*Gs2viga*Bviga*x*(dviga-(x/3));
Esviga1 = fyviga/Esviga;
Fi31 = Esviga1/(dviga-x);
x2 = (areaviga*fyviga)/(0.8*fcviga*Bviga);
M32 = x2*0.8*fcviga*Bviga*(dviga-(0.8*x2/2));

```

```

Fi32 = Esviga1/(dviga-x2);
M3 = max(M31,M32);
M4 = M3;
Fi4 = esuviga/(dviga-x2);
app.MomentoRotura.Value = M4;
app.Orotura.Value = Fi4;

```

%5 Definir valores para lcolumna y Ec1columna

```

lcolumna = ((Bcolumna*(Hcolumna^3))/12);
Ec1columna = fctcolumna/Eccolumna;

```

%6 mostrar cálculo de Momento curvatura columna fisura

```

app.Momentofisurantecolumna.Value = (fctcolumna * lcolumna)/(Hcolumna/2);
app.Ofisurantecolumna.Value = Ec1columna/(Hcolumna/2);

```

%7 Definir valores de , Ec2, xcolumna, M1columna, M2columna, M3columna, M31columna, M32columna, M4columna, Fi1columna, Fi2columna, Fi31columna, Fi32columna, Fi4columna, pcolumna, areacolumna, dcolumna, Gscolumna, ncolumna, Gs2columna, x2, Escolumna1, z, l

```

ncolumna = Escolumna / Eccolumna;
areacolumna = NoVarillasc * pi * (numerovarillac * 0.0254/16)^2;
dcolumna = Hcolumna - recubrimientocolumna;
pcolumna = areacolumna/(Bcolumna * dcolumna);
xcolumna = dcolumna * ncolumna * pcolumna * (-1+
sqrt((1+(2/(ncolumna*pcolumna)))));
M1columna = app.Momentofisurantecolumna.Value;
Fi1columna = app.Ofisurantecolumna.Value;
Gscolumna = (2*M1columna)/(Bcolumna*xcolumna*(dcolumna-(xcolumna/3)));
Ec2columna = Gscolumna/Eccolumna;
Fi2columna = Ec2columna/xcolumna;
Gs2columna = (2*areacolumna*fycolumna)/(Bcolumna*xcolumna);
M31columna = (1/2)*Gs2columna*Bcolumna*xcolumna*(dcolumna-(x/3));
Escolumna1 = fycolumna/Escolumna;
Fi31columna = Escolumna1/(dcolumna-xcolumna);
x2columna = (areacolumna*fycolumna)/(0.8*fccolumna*Bcolumna);
M32columna = x2columna*0.8*fccolumna*Bcolumna*(dcolumna-(0.8*x2columna/2));
Fi32columna = Escolumna1/(dcolumna-x2columna);
M3columna = max(M31columna,M32columna);
M4columna = M3columna;
Fi4columna = esucolumna/(dcolumna-x2columna);

```

```

z = Fi4 + 0.0002;
l = M4+10
app.MomentoRoturacolumna.Value = M4columna;
app.Oroturacolumna.Value = Fi4columna;

```

%8 Calcular Elviga, Elcolumna

```

cla(app.eje,'reset');
datosx = [0,Fi1,Fi32,Fi4];
datosy = [0,M1,M3,M4];
plot(app.eje,datosx,datosy,z,l,'-m');

```

end

% Button pushed function: Irafase2Button

```
function Irafase2ButtonPushed(app, event)
```

```

    Bviga = app.BaseVigam.Value
    Hviga = app.AlturaViga.Value;
    Lviga = app.LongitudViga.Value;
    Bcolumna = app.BaseColumna.Value;
    Hcolumna = app.AlturaColumna.Value;
    Lcolumna = app.LongitudColumna.Value;
    Fi1 = app.Ofisurante.Value;
    Fi2 = app.Orotura.Value;
    Fc = app.fcconcretocolumnaKNm2.Value;
    Ec = app.EcKNm2Columna.Value;
    Tesis2(Bviga,Hviga,Lviga,Bcolumna,Hcolumna,Lcolumna,Fi1,Fi2,Fc,Ec);

```

End

Cortante Basal-Desplazamiento

Código De Programación Un Nivel

% Code that executes after component creation

```
function startupFcn(app, Bviga, Hviga, Lviga, Bcolumna, Hcolumna, Lcolumna, Fi1, Fi2, Fc, Ec)
```

```

    app.BaseColumna.Value = Bcolumna;
    app.BaseViga.Value = Bviga;
    app.Alturaviga.Value = Hviga;
    app.LongitudViga.Value = Lviga;
    app.AlturaColumna.Value = Hcolumna;
    app.LongitudColumna.Value = Lcolumna;
    app.EditField.Value = Fi1;
    app.EditField_2.Value = Fi2;

```



```

app.Fc.Value = Fc;
app.Ec.Value = Ec;
end
% Button pushed function: Button
function ButtonPushed(app, event)
    Bcolumna = app.BaseColumna.Value;
    Bviga = app.BaseViga.Value;
    Hviga = app.Alturaviga.Value;
    Lviga = app.LongitudViga.Value;
    Hcolumna = app.AlturaColumna.Value;
    Lcolumna = app.LongitudColumna.Value;
    Fi1 = app.EditField.Value;
    Fi2 = app.EditField_2.Value;
    Fc = app.Fc.Value;
    Ec = app.Ec.Value;
    Aa = app.Aa.Value;
    Av = app.Av.Value;
    Fa = app.Fa.Value;
    Fv = app.Fv.Value;
    I = app.I.Value;
    GamaConcreto = app.concretoKNm3.Value;
    Sa = app.Sa.Value;
    Periodoaprox = app.Periodoaproximado.Value;
    K = app.K.Value;
    Fx = app.FxKn.Value;
    Fy = app.FyKn.Value;
    Rigidezx = app.RigidezKnM.Value;
    Rigidezx = app.RigidezKnM.Value;
    Desplazamientox = app.DesplazamientoenX.Value;
    Pesoestructura = app.PesodelaestructuraKn.Value;
    Cortantebasal = app.CortantebasalKn.Value;
    Desplazamientoy = app.DesplazamientoenY.Value;
    Desplazamientoductil = app.Desplazamientoductil.Value;
    Desplazamientorotura = app.Desplazamientorotura.Value;
%1 Nombrar Variables lxx, lyy, Sa, Ta, K, w, z,vb,Fix,Fiy,Kx,Ky,Desx,Desy
    lxx = Bcolumna * ((Hcolumna)^3)/12;
    lyy = (Bcolumna^3 * Hcolumna)/12;

```

```

Sa = Aa * Fa * I * 2.5;
Ta = (Lcolumna^0.9) * 0.047
if Ta <= 0.5
K = 1;
elseif Ta > 2.5
K = 2
else
K = 0.75 + 0.5*Ta
end
w = 2*(GamaConcreto * Bcolumna * Hcolumna * (Lcolumna/2)) + (GamaConcreto * Bviga
* Hviga * Lviga);
z = w * (Lcolumna^K);
Vb = w * Sa;
Fix = Vb;
Fiy = Vb;
Kx = 2 * (12 * Ec * Iyy)/(Lcolumna^3);
Ky = 2 * (12 * Ec * Ixx)/(Lcolumna^3);
Desx = Fix/Kx;
Desy = Fiy/Ky;
Fiin = Fi2 - Fi1;
Desrotura = Fiin * Lcolumna
%2 Mostrar Cálculos
app.Sa.Value = Sa;
app.Periodoaproximado.Value = Ta;
app.K.Value = K;
app.CortantebasalKn.Value = Vb;
app.FxKn.Value = Fix;
app.FyKn.Value = Fiy;
app.RigidezexKn.Value = Kx;
app.RigidezenyKn.Value = Ky;
app.DesplazamientoenX.Value = Desx;
app.DesplazamientoenY.Value = Desy;
app.PesodelaestructuraKn.Value = w;
app.Desplazamientoeductil.Value = Desx;
app.Desplazamientorotura.Value = Desrotura;
end

```

Código Programación Dos Niveles**% Code that executes after component creation**

```
function startupFcn(app, Bviga, Hviga, Lviga, Bcolumna, Hcolumna, Lcolumna, Fi1, Fi2, Fc, Ec)
```

```
    app.BaseColumna.Value = Bcolumna;
    app.BaseViga.Value = Bviga;
    app.Alturaviga.Value = Hviga;
    app.LongitudViga.Value = Lviga;
    app.AlturaColumna.Value = Hcolumna;
    app.LongitudColumna.Value = Lcolumna;
    app.EditField.Value = Fi1;
    app.EditField_2.Value = Fi2;
    app.Fc.Value = Fc;
    app.Ec.Value = Ec;
```

```
end
```

% Button pushed function: CalcularButton

```
function CalcularButtonPushed(app, event)
```

```
    Bcolumna = app.BaseColumna.Value;
    Bviga = app.BaseViga.Value;
    Hviga = app.Alturaviga.Value;
    Lviga = app.LongitudViga.Value;
    Hcolumna = app.AlturaColumna.Value;
    Lcolumna = app.LongitudColumna.Value;
    Fi1 = app.EditField.Value;
    Fi2 = app.EditField_2.Value;
    Fc = app.Fc.Value;
    Ec = app.Ec.Value;
    Aa = app.Aa.Value;
    Av = app.Av.Value;
    Fa = app.Fa.Value;
    Fv = app.Fv.Value;
    I = app.I.Value;
    GamaConcreto = app.concretoKNm3.Value;
    Desplazamientoductil = app.Desplazamientoductil.Value;
    Desplazamientorotura = app.Desplazamientorotura.Value;
```

```
%1 Nombrar Variables lxx, lyy, Sa, Ta, K, w, z,vb,Fix,Fiy,Kx,Ky,Desx,Desy,lxx2, lyy2, Sa2, Ta2, K2, w2, z2,vb2,Fix2,Fiy2,Kx2,Ky2,Desx2,Desy2
```

```

Ixx = Bcolumna * ((Hcolumna)^3)/12;
Iyy = (Bcolumna^3 * Hcolumna)/12;
w = 2*(GamaConcreto * Bcolumna * Hcolumna * (Lcolumna)) + (GamaConcreto * Bviga *
Hviga * Lviga);
Kx = 2 * (12 * Ec * Iyy)/(Lcolumna^3);
Ky = 2 * (12 * Ec * Ixx)/(Lcolumna^3);
Fiin = Fi2 - Fi1;
Desrotura = Fiin * Lcolumna
Ixx2 = Bcolumna * ((Hcolumna)^3)/12;
Iyy2 = (Bcolumna^3 * Hcolumna)/12;
Sa2 = Aa * Fa * I * 2.5;
Ta2 = ((2*Lcolumna)^0.9) * 0.047;
if Ta2 <= 0.5
K2 = 1;
elseif Ta > 2.5
K2 = 2
else
K2 = 0.75 + 0.5*Ta
end
w2 = 2*(GamaConcreto * Bcolumna * Hcolumna * (Lcolumna/2)) + (GamaConcreto *
Bviga * Hviga * Lviga);
WT = w + w2;
WHK = w * (Lcolumna^K2);
WHK2 = w2 * (Lcolumna^K2);
WHKT = WHK + WHK2;
CVI1 = WHK/WHKT;
CVI2 = WHK2/WHKT;
VBT = WT * Sa2;
vb1 = VBT * CVI1;
Vb2 = VBT * CVI2;
Fix = vb1;
Fix2 = Vb2;
fix3 = Fix + Fix2
Fiy = vb1;
Fiy2 = Vb2;
F = [Fix;Fix2];
Kx2 = 2 * (12 * Ec * Iyy2)/(Lcolumna^3);

```

```

Ky2 = 2 * (12 * Ec * Ixx2)/(Lcolumna^3);
A = [(Kx2+Kx) (-Kx2); (-Kx2) (Kx2)];
B = inv(A);
D = B * F;
Desx = D(1,1);
Desx2 = D(2,1);
Fiin2 = Fi2 - Fi1;
Desrotura2 = Fiin * (2*Lcolumna);

```

%4 Mostrar Calculos

```

app.PesoprimernivelKn.Value = w;
app.Cortantebasal1Kn.Value = vb1;
app.FxKn.Value = Fix;
app.DesplazamientoenX.Value = Desx;
app.Desplazamientorotura.Value = Desrotura;
app.PesodelaestructuraKn_2.Value = w2;
app.CortantebasalKn_2.Value = Vb2;
app.FxKn_2.Value = Fix2;
app.DesplazamientoenX_2.Value = Desx2;
app.Desplazamientorotura_2.Value = Desrotura2;
app.Sa_2.Value = Sa2;
app.Periodoaproximado_2.Value = Ta2;
app.K_2.Value = Kx2;
app.Cortantebasal.Value = VBT;
app.Pesodelaestructuratotal.Value = WT;
app.Desplazamientoductil_2.Value = Desx2;
app.Desplazamientoductil.Value = Desx;
wl1 = Desrotura2 +0.001;
xl1 = fix3 + 10;

```

%5 Graficar

```

cl1 = [0,Desx,Desx2,Desrotura2];
dl1 = [0,Fix,fix3,fix3];
plot(app.UIAxes,cl1,dl1,wl1,xl1,'r')

```

end

Código De Programación Tres Niveles

% Code that executes after component creation

```

function startupFcn(app, Bviga, Hviga, Lviga, Bcolumna, Hcolumna, Lcolumna, Fi1, Fi2, Fc, Ec)

```

```

app.BaseColumna.Value = Bcolumna;
app.BaseViga.Value = Bviga;
app.Alturaviga.Value = Hviga;
app.LongitudViga.Value = Lviga;
app.AlturaColumna.Value = Hcolumna;
app.LongitudColumna.Value = Lcolumna;
app.EditField.Value = Fi1;
app.EditField_2.Value = Fi2;
app.Fc.Value = Fc;
app.Ec.Value = Ec;
end
% Button pushed function: CalcularButton
function CalcularButtonPushed(app, event)
    Bcolumna = app.BaseColumna.Value;
    Bviga = app.BaseViga.Value;
    Hviga = app.Alturaviga.Value;
    Lviga = app.LongitudViga.Value;
    Hcolumna = app.AlturaColumna.Value;
    Lcolumna = app.LongitudColumna.Value;
    Fi1 = app.EditField.Value;
    Fi2 = app.EditField_2.Value;
    Fc = app.Fc.Value;
    Ec = app.Ec.Value;
    Aa = app.Aa.Value;
    Av = app.Av.Value;
    Fa = app.Fa.Value;
    Fv = app.Fv.Value;
    I = app.I.Value;
    GamaConcreto = app.concretoKNm3.Value;
    Desplazamientoductil = app.Desplazamientoductil.Value;
    Desplazamientorotura = app.Desplazamientorotura.Value;
%1 Nombrar Variables Ixx, Iyy, Sa, Ta, K, w,z,vb,Fix,Fiy,Kx,Ky,Desx,Desy,Ixx2, Iyy2, Sa2,
Ta2, K2, w2, z2,vb2,Fix2,Fiy2,Kx2,Ky2,Desx2,Desy2
    Ixx = Bcolumna * ((Hcolumna)^3)/12;
    Iyy = (Bcolumna^3 * Hcolumna)/12;
    w = 2*(GamaConcreto * Bcolumna * Hcolumna * (Lcolumna)) + (GamaConcreto * Bviga *
    Hviga * Lviga);

```

```

Kx = 2 * (12 * Ec * Iyy)/(Lcolumna^3);
Ky = 2 * (12 * Ec * Ixx)/(Lcolumna^3);
Fiin = Fi2 - Fi1;
Desrotura = Fiin * Lcolumna
Ixx2 = Bcolumna * ((Hcolumna)^3)/12;
Iyy2 = (Bcolumna^3 * Hcolumna)/12;
Sa2 = Aa * Fa * I * 2.5;
Ta2 = ((2*Lcolumna)^0.9) * 0.047;
if Ta2 <= 0.5
K2 = 1;
elseif Ta > 2.5
K2 = 2
else
K2 = 0.75 + 0.5*Ta
end
w2 = 2*(GamaConcreto * Bcolumna * Hcolumna * (Lcolumna)) + (GamaConcreto * Bviga
* Hviga * Lviga);
w3 = 2*(GamaConcreto * Bcolumna * Hcolumna * (Lcolumna/2)) + (GamaConcreto *
Bviga * Hviga * Lviga)
WT = w + w2 + w3;
WHK = w * (Lcolumna^K2);
WHK2 = w2 * (Lcolumna^K2);
WHK3 = w3 * (Lcolumna^K2);
WHKT = WHK + WHK2 + WHK3;
CVI1 = WHK/WHKT;
CVI2 = WHK2/WHKT;
CVI3 = WHK3/WHKT;
VBT = WT * Sa2;
vb1 = VBT * CVI1;
Vb2 = VBT * CVI2;
Vb3 = VBT * CVI3;
Fix = vb1;
Fix2 = Vb2;
Fix3 = Vb3;
fix4 = Fix + Fix2 + Fix3;
Fiy = vb1;
Fiy2 = Vb2;

```

```

F = [Fix;Fix2;Fix3];
Kx2 = 2 * (12 * Ec * Iyy2)/(Lcolumna^3);
Kx3 = 2 * (12 * Ec * Iyy2)/(Lcolumna^3);
Ky2 = 2 * (12 * Ec * Ixx2)/(Lcolumna^3);
A = [(Kx2+Kx) (-Kx2) (0); (-Kx2) (Kx2+Kx3) (-Kx3); (0) (-Kx3) (Kx3)];
B = inv(A);
D = B * F;
Desx = D(1,1);
Desx2 = D(2,1);
Desx3 = D(3,1);
Fiin2 = Fi2 - Fi1;
Desrotura2 = Fiin * (2*Lcolumna);
Desrotura3 = Fiin * (3*Lcolumna);

```

%4 Mostrar Calculos

```

app.PesoprimernivelKn.Value = w;
app.Cortantebasal1Kn.Value = vb1;
app.FxKn.Value = Fix;
app.DesplazamientoenX.Value = Desx;
app.Desplazamientorotura.Value = Desrotura;
app.PesodelaestructuraKn_2.Value = w2;
app.CortantebasalKn_2.Value = Vb2;
app.FxKn_2.Value = Fix2;
app.DesplazamientoenX_2.Value = Desx2;
app.Desplazamientorotura_2.Value = Desrotura2;
app.Sa_2.Value = Sa2;
app.Periodoaproximado_2.Value = Ta2;
app.K_2.Value = Kx2;
app.Cortantebasal.Value = VBT;
app.Pesodelaestructuratotal.Value = WT;
app.Desplazamientoductil_2.Value = Desx2;
app.Desplazamientoductil.Value = Desx3;
app.PesodelaestructuraKn_3.Value = w3;
app.DesplazamientoenX_3.Value = Desx3;
app.Desplazamientoductil_3.Value = Desx3;
app.CortantebasalKn_3.Value = Vb3;
app.DesplazamientoenX_3.Value = Desx3;
app.FxKn_3.Value = Fix3;

```



```
wl1 = Desrotura2 +0.001;  
yl1 = fix4 + 10;  
%3 Graficar  
Dl1 = [0,Desx,Desx2,Desx3,Desrotura3];  
ql1 = [0,Fix,(Fix+Fix2),fix4,fix4];  
plot(app.UIAxes,dl1,ql1, wl1,yl1,'r')  
end
```