بسم الله الرحمن الرحيم

**THE ISLAMIC UNIVERSITY – FACULTY OF ENGINEERING  
COMPUTER ENGINEERING DEPARTMENT**

**SIGNALS AND LINEAR SYSTEMS LABORATORY  
`EELE 3110**

**GUI program Matlab**

***By***

Araba Abed 120191573  
Ahmad Mazen Elhelou 120190203  
Ibrahim Tarek Eslim 120190607

SUPERVISOR  
Eng. Mahmoud H Munia

**Gaza, Palestine**

**10/12/2021**

ABSTRACT

This is an applied project for what we have taken in the Signals and Linear System lab course, using MATLAB and App Designer to create a simple project that give users the ability to do a lot of different mathematical stuff that they need in the signals field, and make the life easier for them.

# Introduction

## About MATLAB:

MATLAB (an abbreviation of "MATrix LABoratory") is a proprietary multi-paradigm programming language and numeric computing environment developed by MathWorks. MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages.

Although MATLAB is intended primarily for numeric computing, an optional toolbox uses the MuPAD symbolic engine allowing access to symbolic computing abilities. An additional package, Simulink, adds graphical multi-domain simulation and model-based design for dynamic and embedded systems.

As of 2020, MATLAB has more than 4 million users worldwide.[21] MATLAB users come from various backgrounds of engineering, science, and economics.[[1]](#footnote-1)

## About App Desinger:

As they mentioned in their website, “App Designer lets you create professional apps without having to be a professional software developer. Drag and drop visual components to lay out the design of your graphical user interface (GUI) and use the integrated editor to quickly program its behavior.”*[[2]](#footnote-2)*

# Project Components:

This project has six screens, every screen has its functionality, we will explain each separately.

## The Main Screen:

This screen has two functions, the first is plotting any given function, the other is making a convolution between two functions and plotting the result.

### As we see above, this screen contains a lot of components.

### Labels:

All labels here have the same reason which is to guide the user on what and where to do.

### Buttons:

We have two buttons here:

* Plot: to plot the given function with shifting and scaling in the first UI Axis.
* Convolution: making the convolution operation between f and h and plot the result in the second UI Axis.

### Text Field:

We have six text field here:

* f(t): read the function from the user, we have to mention the text field read the expression as the MATLAB read, so the user should be aware to use dot product and multiplication to avoid errors.
* h(t): the same as f(t) text field, read h(t), but if the user left it empty, don’t worry it has a default value which is rect(t/2).
* a, b and c: time scaling, time shifting and y scaling respectively.
* Interval: it reads the start, increasing value and the end to make an interval of values to plot them.

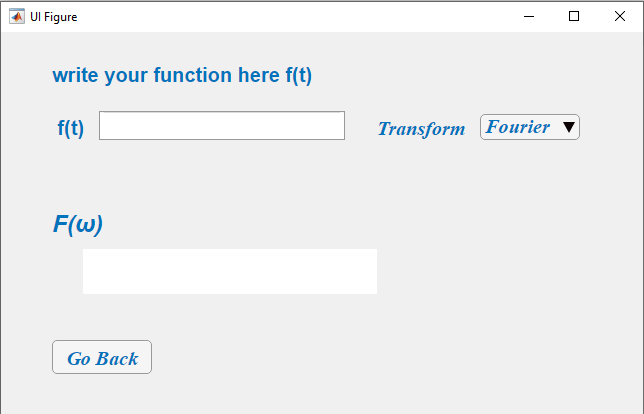
### UI Axis:

We have two of them, the first one to plot the processed f(t) at the given interval, the second plot the result of the convolution.

### Drop Down List:

The most interesting part of our project, each option will open an interesting screen which contains its own benefits.

## The Transformations Screen:

This screen has different transformations (Fourrier , Laplace and Z ).

As we see above, this screen contains a components for transformations.

### Label:

* labels guide the user what and where to do.
* Label to display the transformations result

### Text Field:

f(t): read the function from the user, to convert it from a time domain to the desired domain

### Drop Down List:

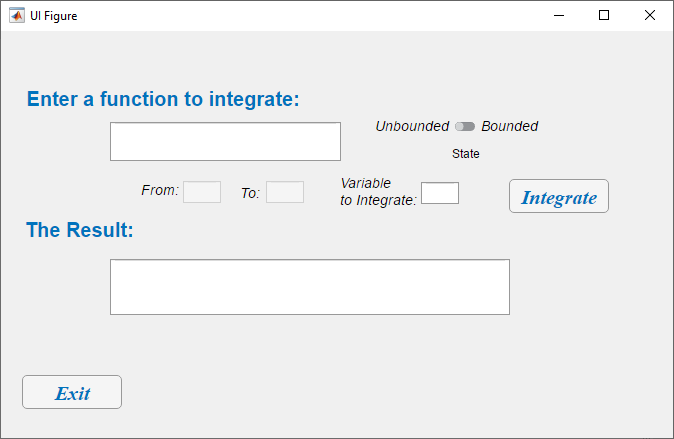
When you click on it, a list of transfers to different domains appear (Fourier, Laplace and Z).

### Buttons:

Go back button: to return main screen.

## The Integral Screen:

This screen can integrate any equation with respect to a specific variable, and the integration can be limited if it gives start and end values, and it can be without limits if you wish.



As we see above, this screen contains a components for integration.

### Labels:

All labels here have the same reason which is to guide the user on what and where to do.

### Text Field:

* read the function from the user, on which you want to perform the integration
* two text fields to get limited integral
* variable to integrate, to get the variable you want to integrate
* text field to display the result of the integration

### Buttons:

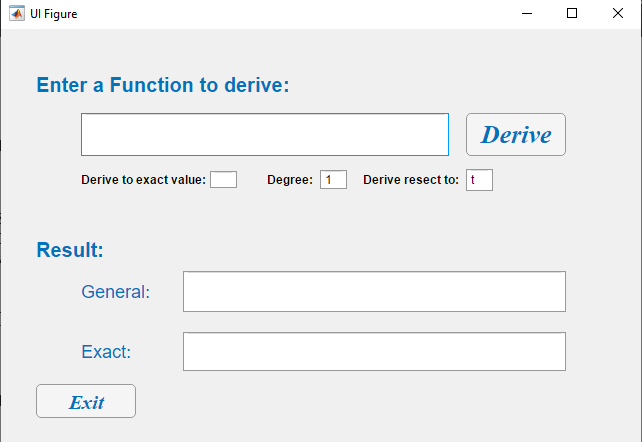
* A button when you press it, the integration process
* Exit button : to exit the integral screen but not close the main screen

### State Switch

This key makes you decide if you want the integration with bounds (from beginning to end) or it will close these fields and the integration will be unlimited

## The Derivative Screen:

In this screen, you can make a derivation for a function that the user enters, and you can specify the variable you want to derive for and the degree of derivation across the fields. You can also when the value of the variable is equal to a certain number



As we see above, this screen contains a components for derivatives.

### Labels:

All labels here have the same reason which is to guide the user on what and where to do.

### Text Field:

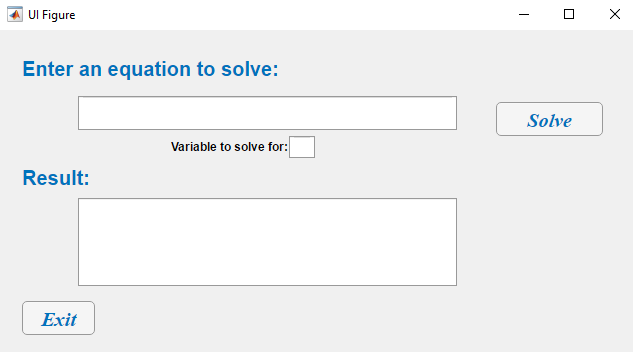
* read the function from the user, on which you want to perform the derivation.
* Text field to Select the degree of derivation you want.
* Text field the variable you want to derive.
* Text field to derive for a given value.
* Tow text field the first is to display the derivation in general, and the second is to display the derivative at the given value.

### Buttons:

* A button when you press it, the derivation process occurs
* Exit button : to exit the derivatives screen but not close the main screen

## The Solve Equations Screen:

This screen asks the user to enter an equation, enter the variable, and solve the equation



As we see above, this screen contains a components for solve equations.

### Labels:

All labels here have the same reason which is to guide the user on what and where to do.

### Text Field:

* Read the function from the user, which you want to solve it.
* Text field The variable you want to solve the equation for it.
* display the result of solving the equation

### Buttons:

* A button when you press it, the solve equations process occurs.
* Exit button : to exit the solve equations screen but not close the main screen.

# Conclusion:

We will type what we learn from this project

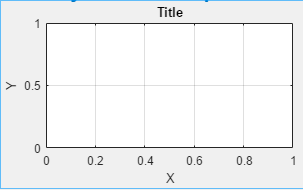
**Ideas project:**

* **Plot the function**
* **Plot the function transformation**
* **Plot the convolution two signal**
* **Integrals**
* **Derivatives**
* **Solve equations**

**Design steps :**

**main screen:**

* **UI-Axes )** **To display the graphic in a screen Convolution or scaled or shift or plot (**

****

* ** Edit Field )** **to write the function(**
* **Label for guidance**
* **Three button :**

1. **Button Convolution**
2. **Button plot**
3. **Button To go to the Transformation screen**

* **Parameter Edit Field and label Group(Specialist in scaling, frequency and offset(**
* **Tools Drop Down (Go to the Transforms, Integrals, Derivatives, and Solve Equations screen(**

**Screen Transformations:**

* ** Edit Field )** **to write the function(**
* **Label :**

1. **Label for guidance**
2. **Label for display the output**

* **Tools Drop Down(to go Fourier, Laplace ,Z )**
* **Button (go back)**

**Screen integrals:**

* Edit field :

1. Edit field to enter the function
2. Two edit field to get integration limits
3. Edit field to get variable to integrate
4. Edit field to display the output

* Label (for guidance)
* Button :

1. Integrate
2. Exit

* StateSwitch (to switch from unbounded to bounded integral limits)

**Screen derivatives :**

* Edit field :

1. Edit field to enter the function
2. Edit field to get degree
3. Edit field to get variable
4. Edit field to find derivative in exact value
5. Edit field to display the output in general form (if don’t put exact value )
6. Edit field to display the output in exact

* Label (for guidance)
* Button :

1. Derivatives
2. Exit

**Scree solve equation:**

* Label (for guidance)
* Edit field :

1. Edit field to enter the function
2. Edit field to get variable you want solve respect with
3. Edit field to display the output

* Button :

1. Solve equation
2. Exit

**Conclusion:**

1. **Learned how to work with Matlab and App designer.**
2. **Designed multi full functional user interface.**
3. **Gained deeper understanding of mathematics concepts.**
4. **Applied different transformation between time and frequency domains.**
5. **Uploading and working with Git and GitHub.**
6. **Working and synchronizing efforts within a team.**
7. **Learned to use different resources, googling skills and thorough debugging of code.**

1. <https://en.wikipedia.org/wiki/MATLAB> [↑](#footnote-ref-1)
2. <https://www.mathworks.com/products/matlab/app-designer.html> [↑](#footnote-ref-2)