

Punto 1:

```
# Método privado con la información de la lista criterio
github-classroom[bot] +1
def __inicializar_criterios(self):

    self.criterios = [Criterio("Desarrollo y profundidad en el tratamiento del tema", 0.2),
                      Criterio("Desafío académico y científico del tema", 0.15),
                      Criterio("Cumplimiento de los objetivos propuestos", 0.1),
                      Criterio("Creatividad e innovación de las soluciones y desarrollos propuestos", 0.1),
                      Criterio("Validez de los resultados y conclusiones", 0.15),
                      Criterio("Manejo y procesamiento de la información y bibliografía", 0.1),
                      Criterio("Calidad y presentación del documento escrito", 0.075),
                      Criterio("Presentación oral", 0.075),
                      Criterio("Nuevo Criterio", 0.05)]
```

Nuevo Criterio

Valor de: 5.0 %

9. Nota Jurado 1

0.00

– +

9. Nota Jurado 2

0.00

– +

9. Observación

Sin Comentarios.

Punto 2: (Es observar el código)

Punto 3:

```
def agregar_acta(st, controlador):
    st.title("Generación De Actas")
    col1, col2, col3 = st.columns(3)
    col5, col6, col7, col8, col9 = st.columns(5)
    # Objeto que modelará el formulario
    info_acta_obj = InfoActa(controlador.criterios)
    info_acta_obj.fecha_acta = datetime.today().strftime('%Y-%m-%d')
    with col1:
        info_acta_obj.autor = st.text_input("Autor")
    with col2:
        info_acta_obj.nombre_trabajo = st.text_input("Nombre De Trabajo")
    with col3:
        info_acta_obj.tipo_trabajo = st.selectbox('Tipo', ('Aplicado', 'Investigación'))
    with col5:
        info_acta_obj.director = st.selectbox("Director", (controlador.directores()[0], controlador.
    with col6:
        info_acta_obj.codirector = st.text_input("Codirector", "N.A")
    with col7:
        info_acta_obj.jurado1 = st.text_input("Jurado #1")
        info_acta_obj.jurado1_1 = st.checkbox("Externo", key="check1")
    with col8:
        info_acta_obj.jurado2 = st.text_input("Jurado #2")
        info_acta_obj.jurado2_2 = st.checkbox("Externo", key="check2")
    with col9:
        info_acta_obj.fecha_presentacion = st.text_input("Fecha de Presentacion")
    enviado_btn = st.button("Enviar")
```

Generación De Actas

Autor	Nombre De Trabajo	Tipo		
<input type="text"/>	<input type="text"/>	Aplicado ▾		
Director	Codirector	Jurado #1	Jurado #2	Fecha de Presentacion
Luisa Ferna... ▾	N.A	<input type="text"/>	<input type="text"/>	<input type="text"/>
		<input type="checkbox"/> Externo	<input type="checkbox"/> Externo	
<input type="button" value="Enviar"/>				

No Deje Ningún Espacio En Blanco En Los Datos

Punto 4:

```
Jose +1
def directores(self):
    directores = ["Luisa Fernanda Rincon", "Juan Carlos Martinez", "Maria Constanza Pabon", "Gloria Inés Álvarez",
                  "Gerardo M Sarria M", "Luis Eduardo Tobón", "Juan Pablo García", "Frank Martinez",
                  "Carlos Ramirez"]

    return directores
```

Punto 5:

```
def agregar_acta(st, controlador):
    st.title("Generación De Actas")
    col1, col2, col3 = st.columns(3)
    col5, col6, col7, col8, col9 = st.columns(5)
    # Objeto que modelará el formulario
    info_acta_obj = InfoActa(controlador.criterios)
    info_acta_obj.fecha_acta = datetime.today().strftime('%Y-%m-%d')
    with col1:
        info_acta_obj.autor = st.text_input("Autor")
    with col2:
        info_acta_obj.nombre_trabajo = st.text_input("Nombre De Trabajo")
    with col3:
        info_acta_obj.tipo_trabajo = st.selectbox('Tipo', ('Aplicado', 'Investigación'))
    with col5:
        info_acta_obj.director = st.selectbox("Director", (controlador.directores()[0], controlador.directores()[1],
    with col6:
        info_acta_obj.codirector = st.text_input("Codirector", "N.A")
    with col7:
        info_acta_obj.jurado1 = st.text_input("Jurado #1")
        info_acta_obj.jurado1_1 = st.checkbox("Externo", key="check1")
    with col8:
        info_acta_obj.jurado2 = st.text_input("Jurado #2")
        info_acta_obj.jurado2_2 = st.checkbox("Externo", key="check2")
    with col9:
        info_acta_obj.fecha_presentacion = st.text_input("Fecha de Presentacion")
    enviado_btn = st.button("Enviar")
```

🔗 Generación De Actas

Autor	Nombre De Trabajo	Tipo		
<input type="text"/>	<input type="text"/>	Aplicado ▼		
Director	Codirector	Jurado #1	Jurado #2	Fecha de Presentacion
Luisa Ferna... ▼	N.A	<input type="text"/>	<input type="text"/>	<input type="text"/>
Luisa Fernan...		<input type="checkbox"/> Externo	<input type="checkbox"/> Externo	
Juan Carlos ...				
Maria Consta...				
Gloria Inés Ál...				
Gerardo M Sa...				
Luis Eduardo ...				
Juan Pablo G...				
Frank Martinez				

espacio En Blanco En Los Datos

Punto 6:

```
class InfoActa:
    # Constructor
    # github-classroom[bot] +1
    def __init__(self, criterios) -> None:
        super().__init__()

        # Datos del acta
        self.autor = ""
        self.fecha_acta = ""
        self.fecha_presentacion = ""
        self.nombre_trabajo = ""
        self.tipo_trabajo = ""
        self.director = ""
        self.codirector = ""
        self.jurado1 = ""
        self.jurado1_1 = False ←
        self.jurado2 = ""
        self.jurado2_2 = False ←
        self.nota_final = 0.0
        self.criterios = criterios
        self.estado = False
```

```
def agregar_acta(st, controlador):
    st.title("Generación De Actas")
    col1, col2, col3 = st.columns(3)
    col5, col6, col7, col8, col9 = st.columns(5)
    # Objeto que modelará el formulario
    info_acta_obj = InfoActa(controlador.criterios)
    info_acta_obj.fecha_acta = datetime.today().strftime('%Y-%m-%d')
    with col1:
        info_acta_obj.autor = st.text_input("Autor")
    with col2:
        info_acta_obj.nombre_trabajo = st.text_input("Nombre De Trabajo")
    with col3:
        info_acta_obj.tipo_trabajo = st.selectbox('Tipo', ('Aplicado', 'Investigación'))
    with col5:
        info_acta_obj.director = st.selectbox("Director", (controlador.directores()[0], controlador.
    with col6:
        info_acta_obj.codirector = st.text_input("Codirector", "N.A")
    with col7:
        info_acta_obj.jurado1 = st.text_input("Jurado #1")
    → info_acta_obj.jurado1_1 = st.checkbox("Externo", key="check1")
    with col8:
        info_acta_obj.jurado2 = st.text_input("Jurado #2")
    → info_acta_obj.jurado2_2 = st.checkbox("Externo", key="check2")
    with col9:
        info_acta_obj.fecha_presentacion = st.text_input("Fecha de Presentacion")
    enviado_btn = st.button("Enviar")
```

Generación De Actas

Autor	Nombre De Trabajo	Tipo		
<input type="text"/>	<input type="text"/>	Aplicado ▼		
Director	Codirector	Jurado #1	Jurado #2	Fecha de Presentacion
Luisa Ferna... ▼	N.A	<input type="text"/>	<input type="text"/>	<input type="text"/>
		<input type="checkbox"/> Externo	<input type="checkbox"/> Externo	
<input type="button" value="Enviar"/>		↑	↑	

No Deje Ningún Espacio En Blanco En Los Datos

Punto 7:

```
def _dibujar_layout(self):
    st.set_page_config(page_title="Actas De Grado", page_icon=':)', layout="wide", initial_sidebar_state="expanded")
    self.col1, self.col2, self.col3, self.col4 = st.columns([1, 1, 1, 1])

    # Definimos lo que abra en la barra de menu
    with st.sidebar:
        st.image("img/LogoJaverianaCali.jpg", width=297)
        self.menu_actual = option_menu("Menu", ["Inicio", 'Crear acta', 'Evaluar acta', 'Exportar acta', 'Estadísticas', 'Ver historicos'],
                                      icons=['house', 'mortarboard', 'people', 'person-check-fill'], menu_icon="cast", default_index=0)
```

```
def controlar_menu(self):
    if self.menu_actual == "Inicio":
        # Se llama con self pq en metodo de la clase MainView
        texto = self.mostrar_bienvenida()
        st.write(texto)
    elif self.menu_actual == "Crear acta":
        # No necesitan self pq son funciones del archivo EvaluacionActaPartial
        agregar_acta(st, self.controller)
    elif self.menu_actual == "Evaluar acta":
        evaluar_criterios(st, self.controller)
    elif self.menu_actual == "Exportar acta":
        exportar_acta(st, self.controller)
    elif self.menu_actual == "Ver históricos":
        ver_historico_acta(st, self.controller)
    elif self.menu_actual == "Estadísticas":
        estadisticas(st, self.controller)
```




Pontificia Universidad
JAVERIANA
Cali



Menu



Inicio



Crear acta



Evaluar acta



Exportar acta



Estadísticas



Ver históricos

Punto 8:

```
def exportar_acta(self, st, controlador, acta_seleccionada):  
    """  
    Exporta el acta en self.pdf  
    :param controlador:  
    :param acta_seleccionada:  
    :return:  
    """  
  
    global criterio  
    numero = 1  
    flag = False  
    self.pdf.add_page()  
    self.pdf.set_font("times", 'B', size=12)  
    self.pdf.image("img/LogoJaverianaCali.jpg", 10, 8, 40)  
    self.pdf.cell(200, 10, txt="ACTA DE EVALUACIÓN DE GRADO", ln=1, align='C')  
    self.pdf.cell(200, 10, txt="Facultad de Ingeniería.", ln=2, align='C')  
    self.pdf.set_font("times", size=12)  
    for acta in controlador.actas:  
        if acta.autor == acta_seleccionada:  
            flag = True  
            → self.pdf.multi_cell(185, 6, txt=f"Trabajo de grado denominado: {acta.nombre_trabajo}", align='L')  
            self.pdf.cell(200, 10, txt=f"Autor: {acta.autor}", ln=1, align='L')  
            self.pdf.cell(200, 10, txt=f"Fecha/Periodo: {acta.fecha_acta} Tesis II", ln=1, align='L')  
            self.pdf.cell(200, 10, txt=f"Director: {acta.director}", ln=1, align='L')  
            self.pdf.cell(200, 10, txt=f"Codirector: {acta.codirector}", ln=1, align='L')  
            self.pdf.cell(200, 10, txt=f"Nombre Del Trabajo: {acta.nombre_trabajo}", ln=1, align='L')  
            self.pdf.cell(200, 10, txt=f"Tipo De Trabajo: {acta.tipo_trabajo}", ln=1, align='L')  
            self.pdf.cell(200, 10, txt=f"Jurado 1: {acta.jurado1}", ln=1, align='L')  
            self.pdf.cell(200, 10, txt=f"Jurado 2: {acta.jurado2}", ln=1, align='L')  
            self.pdf.cell(200, 10, txt=f"Fecha de Presentacion: {acta.fecha_presentacion}", ln=1, align='L')  
            self.pdf.multi_cell(185, 6, txt=f"En atención al desarrollo de este Trabajo de Grado y al documento y sustentación que precede", align='L')  
            for criterio in acta.criterios:
```



ACTA DE EVALUACIÓN DE GRADO

Facultad de Ingeniería.

Trabajo de grado denominado: Ecosistemas Tecnologicos

Autor: Jose Daniel

Fecha/Periodo: 2022-11-07 Tesis II

Director: Juan Carlos Martinez

Codirector: N.A

Punto 9:

```
self.pdf.cell(100, 10, txt=f"{{round(acta.nota_final, 2)}}", ln=0, align='C')
self.pdf.cell(30, 10, txt=f"{{controlador.mostrar_de_numero_a_palabras(acta.nota_final)}}", ln=1, align='C')
self.pdf.cell(95, 10, txt="Números", ln=0, align='C')
self.pdf.cell(35, 10, txt="Letras", ln=1, align='C')
self.pdf.set_font("times", size=12)
self.pdf.multi_cell(185, 7, txt=f"Observación adicional: {{criterio.observacion_adicional}}", align='L')
self.pdf.multi_cell(185, 7, txt=f"Restricciones: {{criterio.restriccion}}", align='L')
self.pdf.multi_cell(185, 5,
```

```
def evaluar_criterios(st, controlador):
    st.title("Evaluación de Criterios")
    flag = False
    num = 1
    temp = 0.0
    opcion = st.selectbox('Elija el autor a calificar', [acta.autor for acta in controlador.actas if not acta.estado])
    st.write("#### Criterios")
    for acta in controlador.actas:
        if acta.autor == opcion:
            flag = True
            for criterio in acta.criterios:
                st.write(criterio.descripcion)
                st.write("Valor de:", criterio.porcentaje * 100, "%")
                nota_jurado1 = st.number_input(str(num) + ". Nota Jurado 1", 0.0, 5.0)
                nota_jurado2 = st.number_input(str(num) + ". Nota Jurado 2", 0.0, 5.0)
                criterio.nota = ((nota_jurado1 + nota_jurado2) / 2) * criterio.porcentaje
                criterio.observacion = st.text_input(str(num) + ". Observación", "Sin Comentarios.")
                temp += criterio.nota
                num += 1
            criterio.observacion_adicional = st.text_input(str(num) + ". Observacion adicional", "Sin Comentarios.")
            num += 1
            criterio.restriccion = st.text_input(str(num) + ". Restriccion", "Sin Comentarios.")
            if temp > 3.5:
                st.write("#### Nota Final", temp, "Acta Aprobada.")
```

10. Observacion adicional

Sin Comentarios.

11. Restriccion

Sin Comentarios.

Nota Final 0.0 Acta Reprobada.

Enviar

Llene Todos Los Campos Vacíos.

Punto 10:

```
42
43     if info_acta_obj.tipo_trabajo == 'Aplicado':
44         controlador.proyectos_aplicados += 1
45     else:
46         controlador.proyectos_investigacion += 1
47     if info_acta_obj.jurado1_1 == False or info_acta_obj.jurado2_2 == False:
48         controlador.jurados_externos += 1
49     if info_acta_obj.jurado1_1 == True or info_acta_obj.jurado2_2 == True:
50         controlador.jurados_internos += 1
51
```


```
138     criterio.restriccion = st.text_input(str(num) + ". Restriccion", "Sin Comentarios.")
139     if temp > 3.5:
140         st.write("### Nota Final", temp, "Acta Aprobada.")
141         if temp > 4.8:
142             controlador.proyectos_mayor_48 += 1
```

```
def estadisticas(st, controlador):
    st.title("Estadísticas generales")


    st.metric("Proyectos de Aplicacion", value=controlador.proyectos_aplicados)
    st.metric("Proyectos de Investigación", value=controlador.proyectos_investigacion)
    st.metric("Proyectos con Jurados Externos", value=controlador.jurados_externos)
    st.metric("Proyectos con Jurados Internos", value=controlador.jurados_internos)
    st.metric("Proyectos Superiores a 4.8", value=controlador.proyectos_mayor_48)
```

github-classroom[bot] +1

```
def controlar_menu(self):  
    if self.menu_actual == "Inicio":  
        # Se llama con self pq en metodo de la clase MainView  
        texto = self.mostrar_bienvenida()  
        st.write(texto)  
    elif self.menu_actual == "Crear acta":  
        # No necesitan self pq son funciones del archivo EvaluacionActaPartial  
        agregar_acta(st, self.controller)  
    elif self.menu_actual == "Evaluar acta":  
        evaluar_criterios(st, self.controller)  
  
    elif self.menu_actual == "Exportar acta":  
        exportar_acta(st, self.controller)  
  
    elif self.menu_actual == "Ver históricos":  
        ver_historico_acta(st, self.controller)  
  
    elif self.menu_actual == "Estadísticas":  
        estadisticas(st, self.controller)
```



```
def _dibujar_layout(self):  
    st.set_page_config(page_title="Actas De Grado", page_icon=':', layout="wide", initial_sidebar_state="expanded")  
    self.col1, self.col2, self.col3, self.col4 = st.columns([1, 1, 1, 1])  
  
    # Definimos lo que abra en la barra de menu  
    with st.sidebar:  
        st.image("img/LogoJaverianaCali.jpg", width=297)  
        self.menu_actual = option_menu("Menu", ["Inicio", 'Crear acta', 'Evaluar acta', 'Exportar acta', 'Estadísticas', 'Ver his  
            icons=['house', 'mortarboard', 'people', 'person-check-fill'], menu_icon="cast", default_1
```





Menu

Inicio

Crear acta

Evaluar acta

Exportar acta

▶ **Estadísticas**

▶ Ver históricos

Estadísticas generales

Proyectos de Aplicacion

1

Proyectos de Investigación

0

Proyectos con Jurados Externos

1

Proyectos con Jurados Internos

1

Proyectos Superiores a 4.8

0

Punto 11:

```
91         align='C')
92         self.pdf.cell(200, 5, txt=f" Firma {acta.jurado1}" Firmas {acta.jurado2} " ", ln=2,
93         align='C')
94
```

Firma Juan

Firma Pepe

Punto 12:

```
def estadisticas(st, controlador):  
    st.title("Estadísticas generales")  
  
    st.metric("Proyectos de Aplicación", value=controlador.proyectos_aplicados)  
    st.metric("Proyectos de Investigación", value=controlador.proyectos_investigacion)  
    st.metric("Proyectos con Jurados Externos", value=controlador.jurados_externos)  
    st.metric("Proyectos con Jurados Internos", value=controlador.jurados_internos)  
    st.metric("Proyectos Superiores a 4.8", value=controlador.proyectos_mayor_48)  
  
    st.title("Estadísticas con Plotly")  
  
    labels = ['Proyectos de Aplicación', 'Proyectos de Investigación', 'Proyectos con Jurados Externos', 'Proyectos con Jurados Internos', 'Proyectos Superiores a 4.8']  
    values = [controlador.proyectos_aplicados, controlador.proyectos_investigacion, controlador.jurados_externos, controlador.jurados_internos, controlador.proyectos_mayor_48]  
  
    fig = go.Figure(data=[go.Pie(labels=labels, values=values)])  
    st.plotly_chart(fig, use_container_width=True)
```



Estadísticas generales

Proyectos de Aplicación

1

Proyectos de Investigación

0

Proyectos con Jurados Externos

1

Proyectos con Jurados Internos

0

Proyectos Superiores a 4.8

0

Estadísticas con Plotly

