**Formulario I - Datos Generales de la Beca**:

**-Datos Personales**

◦ Apellidos

◦ Nombres

◦ Género (en el que se autopercibe)

◦ Documento: DNI Nº

◦ Lugar de Nacimiento: País:

◦ Fecha de Nacimiento

◦Dirección Particular: Calle, Nº, Piso, Depto, Localidad, CP, Partido

◦Teléfono: - Celular: -15

◦Correo Electrónico

◦Nº de CUIL/CUIT

◦Título Universitario

◦Universidad de Graduación

◦Fecha de Graduación

◦Fecha de Ingreso

◦Promedio con Aplazos

◦Promedio sin Aplazos

**-Si el/la postulante NO está graduado/a**: (no obligatorio)

◦Fecha Estimativa de Graduación

◦Cantidad de Materias que Adeuda

-**Datos Relativos a la propuesta de postulación**:

◦Título del tema de Investigación propuesto

◦Palabras claves del plan (tres palabras claves en mayúsculas)

-**Líneas de la convocatoria**: (seleccionar entre Centro Propio, Línea priorizada y línea abierta. Es obligatorio)

(depende de la selección deberá seleccionar según lo que corresponda):

▪Línea Centro Propios (Seleccionar entre: CEREN, LAL, LINTA, CPP, CEMECA, LEMIT, CESGI. Son las siglas de los centros propios)

▪Línea Priorizada (Seleccionar entre: Salud, Industria y Producción, Ambiente, Desarrollo a la comunidad)

▪Línea Abierta (describir) Permitir hasta 66 caracteres.

**-Comisión Asesora Honoraria a la que se presenta**: (Seleccionar una del siguiente listado: Ciencias Agrícolas, Producción y Salud Animal; Ciencias Biológicas y Salud; Ciencias Sociales y Humanas; Física, Matemática, Química, Astronomía; Geología. Minería e Hidrología; Ingeniería, Arquitectura y Tecnología; TIC'S, Electrónica e Informática; Ciencias Ambientales y Territorio)

-**Director/a de la Beca**...:

◦Apellido

◦Nombres

◦Título Máximo

◦Lugar de trabajo y realización de la beca

◦Cantidad de Becarios que dirige (máximo 5)

◦Dirección de correo electrónico

◦T.E. (indicar interno)

◦Cargos:

▪¿Pertenece a la Carrera del Investigador CIC o CONICET? Indicar categoría.

▪¿Tiene el cargo de Profesor Adjunto (o superior) y con una dedicación Semi Exclusiva (o superior)? Indicar cargo y dedicación.

▪¿Tiene la categoría I, II o III en el Programa de Incentivos?

▪¿Posee una categoría equivalente en instituciones del sistema científico- tecnológico con lugar de trabajo en la provincia de Buenos Aires? Indicar cargo de planta que reviste en esa institución.

•**Co-Director/a de la Beca (No obligatorio)**:

◦Apellido

◦Nombres

◦Título Máximo

◦Lugar de trabajo y realización de la beca

◦Cantidad de Becarios que dirige (máximo 5)

◦Dirección de correo electrónico

◦T.E.: (indicar interno)

◦Cargos:

▪¿Pertenece a la Carrera del Investigador CIC o CONICET? Indicar categoría.

▪¿Tiene el cargo de Profesor Adjunto (o superior) y con una dedicación Semi Exclusiva (o superior)? Indicar cargo y dedicación.

▪¿Tiene la categoría I, II o III en el Programa de Incentivos?

▪¿Posee una categoría equivalente en instituciones del sistema científico- tecnológico con lugar de trabajo en la provincia de Buenos Aires? Indicar cargo de planta que reviste en esa institución.

-**Lugar propuesto para desarrollar la Beca**:(acá tiene que tener al menos un campo completo entre centro, Universidad, Facultar y otros. Quiero poner una lista desplegable con los centros y universidades y facultades de la base de datos)

◦Centro

◦Universidad

◦Facultad

◦Departamento

◦Cátedra

◦Instituto

◦Otros

-Tipo de asociación con la CIC ( Acá seleccionar una de estas opciones: Centro Propio, Centro de Asociación Múltiple, Centro de Asociación Simple, Sin Asociación/otros)

◦Dirección del Lugar: Calle, Nº, Localidad, C.P.

◦Correo electrónico

◦T.E.

**Formulario II – Plan y Lugar de Trabajo**:

•**Del plan de Beca:**

1.Denominación del trabajo

2.Definición del problema y estado actual del conocimiento sobre la cuestión

3.Trabajo previo realizado referente a este proyecto

4.Objetivo/s general/es y objetivos particulares.

5.Métodos y técnicas a emplear.

6.Cronograma mensual de actividades a desarrollar en el período de la beca (comprendido por un período de 12 meses).

7.Planificación a 4 años del plan de trabajo.

8.Referencias bibliográficas.

9.Vinculación del plan de trabajo con otros proyectos de investigación en ejecución en el mismo lugar de trabajo.

10.¿El abordaje del plan de trabajo está enmarcado en alguna problemática de la provincia de Buenos Aires? Justificar

•**Del lugar de trabajo propuesto para el desarrollo de la Beca**14...:

1.Identificación del lugar donde se realizará el plan de trabajo

2.Descripción de la infraestructura y servicios disponibles en relación a los requerimientos del plan de trabajo.

Repasando los requerimientos:

Contexto: la Comision de investigaciones Científicas de la provincia de buenos aires (CIC), entre otras políticas publicas, lanza aperturas para la inscripción de concursos para el otorgamiento de becas, subsidios o para el ingreso a la planta en carreras de investigador o personal de apoyo. Para presentarse a estos, los postulantes realizan las presentaciones adjuntado documentación: Formularios con datos, DDJJ, títulos de estudios, CVs, etc.

Este sistema tiene como fin soportar la confección de los formularios que contienen datos importantes para procesar y seleccionar las presentaciones durante las etapas del concurso. Actualmente se realiza a través de archivos editables sin ningún tipo de validaciones. Esto implica que la mayoría de la veces hay datos faltantes o no están ingresados de manera correcta. Esto implica un desgaste y un proceso engorroso simplemente para asegurar que estén bien enviados los formularios. También, una vez cerrado el concurso se vuelcan en un Excel todos los datos de las presentaciones, quedando un archivo consolidado para luego ser trabajado por los equipos de administración. Esta propuesta también es para que los datos sean almacenados cuando el postulante carga los datos, a si las personas que trabajan en la adminsitracion pueden descargar el Excel con todos los datos, ahorrando tiempo y disminuyendo los error involuntarios por el traspaso de datos.

Hay dos usuarios, el usuario rol postulante (externo) y el usuario rol administrador.

Usuario postulante:

Este usuario ingresa a la pagina de la CIC, https://www.cic.gba.gob.ar/cic-convocatorias/, en el cual siempre se cuelgan las novedades y los concursos abiertos. Aquí va a encontrar todas las documentaciones y requisitos necesarios para que pueda hacer la presentación. Parte de la documentación son, en general, dos formularios (datos personales y datos relacionados al lugar de trabajo por ejemplo para las becas). Este usuario va a poder acceder a la confección de formularios, lo cual antes se deberá registrar (el usuario tiene que ser su correo electrónico). Una vez registrado, inicia sesión y visualizara los concursos que están activos para los cuales se puede postular. Selecciona uno y le aparece el formulario I para la carga, una vez finalizada la carga hay un botón que indica seguir con formulario II, y comienza la carga del segundo formulario. Una vez finalizado presiona el botón finalizar. Al presionar finalizar, va a poder generar el formulario en PDF (formato aun a definir pero por ahora como el que esta compartido). Estaría bueno que el formulario tenga algún código de validación. También hay que contemplar que se puedan ir guardando los datos que vaya cargando para que se pueda ir completando en varias instancias antes de que sea finalizado. El usuario también debe poder ver los formularios generados.

Usuario Rol administrador:

Este usuario es quien podrá generar los concursos para que se visualicen, editar y definir los formularios, descargar en un Excel la tabla de los postulantes de un determinado concurso.

Existen 8 tipos básicos de concursos, Concurso para ingreso como investigador (CICINV), para ingreso como personal de apoyo a la investigación (CICPA), para ingreso como becario de entrenamiento (BENTRE), para ingreso como becario doctoral (BEDOC), para subsidios de publicación (PDCT), para subsidio de organización (ORCT), para subsidio de Asistencia (ARCT) y subsidio idea proyecto (IP). En general estos grupos tienen casi la misma estructura de datos, varia según el tipo, y de una convocatoria a otra puede ser que se modifiquen cosas. Es por eso que debemos tener formularios para cada tipo de subsidio, con la posibilidad que el usuario administrador pueda editar los campos a completar.

Usuario Postulante:

Inicia secion o se registra si no esta registrado.

* Al ingresar vera un panel donde tiene opciones, ver los concursos en los cuales ya genero los formularios para presentar o aquellos que no están finalizados, que están en estado sin finalizar. Y si quiere podrá descargar los formularios en pdf de esas postulaciones. La otra opción es que puede seleccionar a algún concurso activo para confeccionar los formularios de presentacion. Presionando esta opción, se desplega los concursos activos. Pero Para los cuales ya genero un formulario, le saldrá un cartel informando que ya esta el formulario presentado.
* En la confecion de formularios, el postulante podera ir cargándolos, guardándolos datos cargados para que lo peda completar en etapas (ya que es muy largo). Para luego ingresar y poder seguir completando hasta generarlo.

Usuario Administrador:

* Da de alta a otros administradores.
* Genera concursos activos
* Genera los formularios para cada concurso (ya que estos varían en tipo: BEDOC, BENTRE, CICPA, CICINV, etc; y pueden variar, dentro del mismo tipo algún dato que se pida).
* Descargar en un Excel todos un consolidado con todos los datos cargados de un determinado concurso.
* También podrá visualizar un tablero con indicadores de seguimiento de concurso, por ejempli, cuantas postulaciones hay, cuantas con formulario finalizado y cuantas sin finalizar, etc.

Se un experto pompt y experto en sistemas, y audita los soguientes arhivos de funcioalidad del sistema. Para detectar errores y mejoras. Se muy profecional y exigente. Al terminar, redactamen a si compio y pego las indicaciones que le tengo que dar a chatgpt para que las corrija:

App.py

# rutas\_postulante.py

from flask import Blueprint, render\_template, request, redirect, url\_for, flash, jsonify, send\_file

from flask\_login import login\_required, current\_user

from extensions import db

from models import Concurso, FormularioBase, CampoFormulario, RespuestaCampoDinamico

import json

import pdfkit

import os

postulante\_bp = Blueprint('postulante', \_\_name\_\_)

# ----------------- TABLERO POSTULANTE -----------------

@postulante\_bp.route("/panel\_postulante", methods=["GET"])

@login\_required

def panel\_postulante():

if current\_user.role != "postulante":

flash("Acceso denegado.", "danger")

return redirect(url\_for("auth.login\_postulante"))

postulaciones\_borrador = FormularioBase.query.filter\_by(postulante\_id=current\_user.id, estado="borrador").all()

postulaciones\_finalizadas = FormularioBase.query.filter\_by(postulante\_id=current\_user.id, estado="finalizado").all()

concursos\_abiertos = Concurso.query.filter(Concurso.estado == "abierto").all()

return render\_template(

"panel\_postulante.html",

postulaciones\_borrador=postulaciones\_borrador,

postulaciones\_finalizadas=postulaciones\_finalizadas,

concursos\_abiertos=concursos\_abiertos

)

# ----------------- LISTAR CONCURSOS ACTIVOS -----------------

@postulante\_bp.route("/concursos\_activos", methods=["GET"])

@login\_required

def concursos\_activos():

if current\_user.role != "postulante":

flash("Acceso denegado.", "danger")

return redirect(url\_for("auth.login\_postulante"))

concursos = Concurso.query.filter\_by(estado="abierto").all()

return render\_template("concursos\_activos.html", concursos=concursos)

# ----------------- INICIAR FORMULARIO -----------------

@postulante\_bp.route("/completar\_formulario/<int:concurso\_id>", methods=["GET", "POST"])

@login\_required

def completar\_formulario(concurso\_id):

if current\_user.role != "postulante":

flash("Acceso denegado.", "danger")

return redirect(url\_for("auth.login\_postulante"))

concurso = Concurso.query.get\_or\_404(concurso\_id)

formulario = FormularioBase.query.filter\_by(postulante\_id=current\_user.id, concurso\_id=concurso\_id).first()

campos = CampoFormulario.query.filter\_by(concurso\_id=concurso\_id).all()

if formulario:

if formulario.estado == "finalizado":

flash("Ya has completado este formulario. No puedes postularte dos veces.", "warning")

return redirect(url\_for("postulante.panel\_postulante"))

if not campos:

flash("No hay campos definidos para este concurso. Contacte al administrador.", "warning")

return redirect(url\_for("postulante.panel\_postulante"))

if request.method == "POST":

datos\_formulario = json.dumps(request.form.to\_dict())

if formulario:

formulario.datos\_generales = datos\_formulario

else:

formulario = FormularioBase(

postulante\_id=current\_user.id,

concurso\_id=concurso\_id,

datos\_generales=datos\_formulario,

estado="borrador"

)

db.session.add(formulario)

db.session.commit()

flash("Formulario guardado correctamente.", "success")

return redirect(url\_for("postulante.panel\_postulante"))

return render\_template("formulario\_dinamico.html", concurso=concurso, campos=campos)

# ----------------- FINALIZAR FORMULARIO -----------------

@postulante\_bp.route("/finalizar\_formulario/<int:concurso\_id>", methods=["POST"])

@login\_required

def finalizar\_formulario(concurso\_id):

formulario = FormularioBase.query.filter\_by(postulante\_id=current\_user.id, concurso\_id=concurso\_id).first()

if not formulario:

flash("No hay un formulario guardado para este concurso.", "danger")

return redirect(url\_for("postulante.completar\_formulario", concurso\_id=concurso\_id))

formulario.estado = "finalizado"

db.session.commit()

flash("Formulario enviado correctamente.", "success")

return redirect(url\_for("postulaciones\_enviadas"))

# ----------------- LISTAR FORMULARIOS ENVIADOS -----------------

@postulante\_bp.route("/postulaciones\_enviadas", methods=["GET"])

@login\_required

def postulaciones\_enviadas():

postulaciones = FormularioBase.query.filter\_by(postulante\_id=current\_user.id, estado="finalizado").all()

return render\_template("postulaciones\_enviadas.html", postulaciones=postulaciones)

# ----------------- GENERAR PDF -----------------

@postulante\_bp.route("/generar\_pdf/<int:formulario\_id>", methods=["GET"])

@login\_required

def generar\_pdf(formulario\_id):

formulario = FormularioBase.query.get\_or\_404(formulario\_id)

datos\_generales = json.loads(formulario.datos\_generales)

html = render\_template("postulacion\_pdf.html", datos\_generales=datos\_generales)

pdf\_path = f"postulacion\_{formulario\_id}.pdf"

pdfkit.from\_string(html, pdf\_path)

response = send\_file(pdf\_path, as\_attachment=True)

os.remove(pdf\_path)

return response