

Modelos de Usuarios y Autenticación

```
class User(AbstractUser):
    TIPO_USUARIO = [
        ('admin', 'Administrador'),
        ('medico', 'Médico'),
        ('paciente', 'Paciente'),
        ('empresa', 'Empresa'),
    ]

    tipo_usuario = models.CharField(max_length=10, choices=TIPO_USUARIO)
    empresa = models.ForeignKey('Empresa', on_delete=models.CASCADE, null=True,
blank=True)
    telefono = models.CharField(max_length=15)
    fecha_registro = models.DateTimeField(auto_now_add=True)

class PerfilMedico(models.Model):
    usuario = models.OneToOneField(User, on_delete=models.CASCADE)
    especialidad = models.CharField(max_length=100)
    cedula_profesional = models.CharField(max_length=20)
    horario_disponible = models.JSONField() # {dias: [], horarios: []}

class Empresa(models.Model):
    nombre = models.CharField(max_length=200)
    rfc = models.CharField(max_length=13)
    direccion = models.TextField()
    telefono = models.CharField(max_length=15)
    numero_trabajadores = models.IntegerField()
```

Modelos para Cuestionarios NOM-035

```
class CuestionarioNOM035(models.Model):
    ESTADO_CUESTIONARIO = [
        ('borrador', 'Borrador'),
        ('activo', 'Activo'),
        ('inactivo', 'Inactivo'),
    ]

    nombre = models.CharField(max_length=200)
    version = models.CharField(max_length=10)
    descripcion = models.TextField()
    estado = models.CharField(max_length=10, choices=ESTADO_CUESTIONARIO)
```

```
fecha_creacion = models.DateTimeField(auto_now_add=True)
creado_por = models.ForeignKey(User, on_delete=models.CASCADE)
```

```
class DominioNOM035(models.Model):
    cuestionario = models.ForeignKey(CuestionarioNOM035, on_delete=models.CASCADE)
    nombre = models.CharField(max_length=100)
    descripcion = models.TextField()
    orden = models.IntegerField()
```

```
class PreguntaNOM035(models.Model):
```

```
    TIPO_PREGUNTA = [
        ('likert', 'Escala Likert'),
        ('multiple', 'Opción Múltiple'),
        ('abierta', 'Respuesta Abierta'),
    ]
```

```
    dominio = models.ForeignKey(DominioNOM035, on_delete=models.CASCADE)
    texto = models.TextField()
    tipo = models.CharField(max_length=10, choices=TIPO_PREGUNTA)
    orden = models.IntegerField()
    obligatoria = models.BooleanField(default=True)
```

```
class OpcionRespuesta(models.Model):
```

```
    pregunta = models.ForeignKey(PreguntaNOM035, on_delete=models.CASCADE)
    texto = models.CharField(max_length=200)
    valor = models.IntegerField()
    orden = models.IntegerField()
```

Modelos para Evaluación de Factores de Riesgo Psicosocial

```
class EvaluacionRiesgoPsicosocial(models.Model):
```

```
    empresa = models.ForeignKey(Empresa, on_delete=models.CASCADE)
    cuestionario = models.ForeignKey(CuestionarioNOM035, on_delete=models.CASCADE)
    fecha_inicio = models.DateTimeField()
    fecha_fin = models.DateTimeField()
    estado = models.CharField(max_length=20)
```

```
class RespuestaEvaluacion(models.Model):
```

```
    evaluacion = models.ForeignKey(EvaluacionRiesgoPsicosocial,
on_delete=models.CASCADE)
    empleado = models.ForeignKey(User, on_delete=models.CASCADE)
    pregunta = models.ForeignKey(PreguntaNOM035, on_delete=models.CASCADE)
    respuesta = models.JSONField() # Almacena diferentes tipos de respuestas
    fecha_respuesta = models.DateTimeField(auto_now_add=True)
```

```

class ResultadoEvaluacion(models.Model):
    evaluacion = models.ForeignKey(EvaluacionRiesgoPsicosocial,
on_delete=models.CASCADE)
    empleado = models.ForeignKey(User, on_delete=models.CASCADE)
    puntaje_total = models.FloatField()
    nivel_riesgo = models.CharField(max_length=50) # Bajo, Medio, Alto
    recomendaciones = models.TextField()
    fecha_calculo = models.DateTimeField(auto_now_add=True)

```

Modelos para ATS (Accidente de Trabajo y Salud)

```

class ReporteATS(models.Model):
    TIPO_INCIDENTE = [
        ('accidente', 'Accidente de Trabajo'),
        ('enfermedad', 'Enfermedad Laboral'),
        ('incidente', 'Incidente'),
    ]

    empresa = models.ForeignKey(Empresa, on_delete=models.CASCADE)
    empleado = models.ForeignKey(User, on_delete=models.CASCADE)
    tipo_incidente = models.CharField(max_length=20, choices=TIPO_INCIDENTE)
    fecha_incidente = models.DateTimeField()
    descripcion = models.TextField()
    medidas_tomadas = models.TextField()
    severidad = models.CharField(max_length=20)
    estado = models.CharField(max_length=20)

```

Modelos para Pruebas Psicométricas

```

class PruebaPsicometrica(models.Model):
    CATEGORIA_PRUEBA = [
        ('personalidad', 'Personalidad'),
        ('aptitud', 'Aptitud'),
        ('inteligencia', 'Inteligencia'),
    ]

    nombre = models.CharField(max_length=200)
    categoria = models.CharField(max_length=20, choices=CATEGORIA_PRUEBA)
    descripcion = models.TextField()
    tiempo_limite = models.IntegerField() # en minutos
    activa = models.BooleanField(default=True)

```

```

class PreguntaPsicometrica(models.Model):
    prueba = models.ForeignKey(PruebaPsicometrica, on_delete=models.CASCADE)
    texto = models.TextField()
    imagen = models.ImageField(upload_to='psicometricas/', null=True, blank=True)
    orden = models.IntegerField()

class OpcionPsicometrica(models.Model):
    pregunta = models.ForeignKey(PreguntaPsicometrica, on_delete=models.CASCADE)
    texto = models.CharField(max_length=200)
    valor = models.IntegerField()

class ResultadoPsicometrica(models.Model):
    empleado = models.ForeignKey(User, on_delete=models.CASCADE)
    prueba = models.ForeignKey(PruebaPsicometrica, on_delete=models.CASCADE)
    puntaje = models.FloatField()
    interpretacion = models.TextField()
    fecha_aplicacion = models.DateTimeField(auto_now_add=True)

```

Modelos para Agenda de Citas

```

class CitaMedica(models.Model):
    ESTADO_CITA = [
        ('programada', 'Programada'),
        ('confirmada', 'Confirmada'),
        ('completada', 'Completada'),
        ('cancelada', 'Cancelada'),
    ]

    paciente = models.ForeignKey(User, on_delete=models.CASCADE,
related_name='citas_paciente')
    medico = models.ForeignKey(User, on_delete=models.CASCADE,
related_name='citas_medico')
    fecha_hora = models.DateTimeField()
    duracion = models.IntegerField(default=30) # en minutos
    motivo = models.TextField()
    estado = models.CharField(max_length=20, choices=ESTADO_CITA,
default='programada')
    notas_medico = models.TextField(blank=True)

class DisponibilidadMedico(models.Model):
    medico = models.ForeignKey(User, on_delete=models.CASCADE)
    dia_semana = models.IntegerField() # 1-7 (Lunes-Domingo)
    hora_inicio = models.TimeField()
    hora_fin = models.TimeField()
    activo = models.BooleanField(default=True)

```

Modelos para Reportes y Analytics

```
class ReporteEmpresa(models.Model):
    empresa = models.ForeignKey(Empresa, on_delete=models.CASCADE)
    tipo_reporte = models.CharField(max_length=50)
    periodo = models.CharField(max_length=20)
    datos = models.JSONField()
    fecha_generacion = models.DateTimeField(auto_now_add=True)
```

```
class Notificacion(models.Model):
    usuario = models.ForeignKey(User, on_delete=models.CASCADE)
    titulo = models.CharField(max_length=200)
    mensaje = models.TextField()
    leida = models.BooleanField(default=False)
    fecha_creacion = models.DateTimeField(auto_now_add=True)
    tipo = models.CharField(max_length=20)
```

Modelos para el Sitio Web (Blog)

```
class BlogPost(models.Model):
    titulo = models.CharField(max_length=200)
    slug = models.SlugField(unique=True)
    contenido = models.TextField()
    autor = models.ForeignKey(User, on_delete=models.CASCADE)
    fecha_publicacion = models.DateTimeField()
    imagen_portada = models.ImageField(upload_to='blog/')
    activo = models.BooleanField(default=True)
```

```
class Contacto(models.Model):
    nombre = models.CharField(max_length=100)
    email = models.EmailField()
    telefono = models.CharField(max_length=15)
    mensaje = models.TextField()
    fecha_contacto = models.DateTimeField(auto_now_add=True)
    respondido = models.BooleanField(default=False)
```