
DOCUMENTACIÓN TÉCNICA

TABLA ESTIMATES

Base de Datos JobNimbus Stamford

Tabla:	jobnimbus_estimates
Instancia:	Stamford
Total de Campos:	35+
Campos JSONB:	items, sections, owners, related, custom_fields
Tipo:	Tabla Transaccional
Fecha:	13/11/2025

1. RESUMEN EJECUTIVO

La tabla **jobnimbus_estimates** es fundamental en el proceso de ventas de JobNimbus Stamford. Almacena todas las cotizaciones o presupuestos generados para clientes, incluyendo líneas de productos/servicios, cálculos financieros, estados de aprobación y relaciones con jobs y contactos. La tabla contiene **35+ campos** estructurados más campos JSONB flexibles para ítems y secciones.

Métrica	Valor	Descripción
Total de Campos	35+	Campos estructurados y JSONB
Campos JSONB	5	items, sections, owners, related, custom_fields
Índices	8	Para optimización de consultas
Campos Financieros	5	subtotal, tax, total, cost, margin
Campos de Fecha	5	created, updated, estimate, sent, approved

2. ESTRUCTURA DE LA TABLA

A continuación se presenta la definición SQL completa de la tabla `jobnimbus_estimates`, incluyendo todos los campos, tipos de datos y constraints.

```
CREATE TABLE IF NOT EXISTS jobnimbus_estimates (  
  id UUID PRIMARY KEY DEFAULT uuid_generate_v4(),  
  instance VARCHAR(20) NOT NULL,  
  
  -- Identificación  
  jnid VARCHAR(100) NOT NULL,  
  recid BIGINT,  
  number VARCHAR(50),  
  guid VARCHAR(100),  
  
  -- Fechas (Unix timestamps)  
  date_created BIGINT NOT NULL,  
  date_updated BIGINT,  
  date_estimate BIGINT,  
  date_sent BIGINT,  
  date_approved BIGINT,  
  
  -- Estado  
  status BIGINT,  
  status_name VARCHAR(100),  
  esigned BOOLEAN DEFAULT FALSE,  
  
  -- Montos financieros  
  subtotal DECIMAL(12,2),  
  tax DECIMAL(12,2),  
  total DECIMAL(12,2),  
  cost DECIMAL(12,2),  
  margin DECIMAL(12,2),  
  
  -- Items y secciones (JSONB)  
  items JSONB,           -- Array de items del estimate  
  sections JSONB,        -- Array de secciones  
  
  -- Sales rep y ownership  
  sales_rep_jnid VARCHAR(100),  
  sales_rep_name VARCHAR(255),  
  owners JSONB,  
  
  -- Relaciones  
  related JSONB,          -- Array: jobs, contacts, etc.  
  
  -- Notas  
  note TEXT,  
  internal_note TEXT,  
  terms TEXT,  
  
  -- Template y location
```

```
template_id VARCHAR(100),
location_id BIGINT,

-- Custom fields
custom_fields JSONB,

-- Metadata
is_active BOOLEAN DEFAULT TRUE,
is_archived BOOLEAN DEFAULT FALSE,
created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

UNIQUE(instance, jnid)
);
```

3. CATÁLOGO COMPLETO DE CAMPOS

3.1 Campos de Identificación

Campos únicos que identifican cada estimate en el sistema

Campo	Tipo	Descripción	Constraints
id	UUID	Identificador único interno de PostgreSQL	PRIMARY KEY
instance	VARCHAR(20)	Identificador de instancia multi-tenant (stamford/guilford)	NOT NULL
jnid	VARCHAR(100)	JobNimbus ID único del estimate	NOT NULL, UNIQUE
recid	BIGINT	Record ID numérico de JobNimbus	
number	VARCHAR(50)	Número de estimate (ej: 1902, EST-2025-001)	INDEXED
guid	VARCHAR(100)	Global Unique Identifier	

3.2 Fechas (Unix Timestamps)

Todos los campos de fecha almacenados como timestamps Unix (segundos desde 1970)

Campo	Tipo	Descripción	Constraints
date_created	BIGINT	Fecha de creación del estimate	NOT NULL, INDEXED
date_updated	BIGINT	Fecha de última actualización	
date_estimate	BIGINT	Fecha del estimate (fecha efectiva)	
date_sent	BIGINT	Fecha en que se envió al cliente	
date_approved	BIGINT	Fecha de aprobación por el cliente	INDEXED

3.3 Estado y Firma

Campos que controlan el estado y proceso de firma del estimate

Campo	Tipo	Descripción	Constraints
status	BIGINT	Código numérico de estado (1=Draft, 2=Sent, 4=Approved)	INDEXED
status_name	VARCHAR(100)	Nombre del estado (Draft, Sent, Approved, Rejected)	
esigned	BOOLEAN	Indica si el estimate fue firmado electrónicamente	DEFAULT FALSE

3.4 Montos Financieros

Campos que almacenan valores monetarios y cálculos financieros

Campo	Tipo	Descripción	Constraints
subtotal	DECIMAL(12,2)	Subtotal antes de impuestos	
tax	DECIMAL(12,2)	Monto de impuestos aplicados	
total	DECIMAL(12,2)	Total final (subtotal + tax)	
cost	DECIMAL(12,2)	Costo total de los materiales/servicios	
margin	DECIMAL(12,2)	Margen de ganancia en porcentaje	

3.5 Items y Secciones (JSONB)

Campos JSONB que almacenan los items del estimate y su organización

Campo	Tipo	Descripción	Constraints
items	JSONB	Array de items/productos del estimate	GIN INDEX
sections	JSONB	Array de secciones para organizar items	

3.6 Sales Rep y Ownership

Información del representante de ventas y propietarios del estimate

Campo	Tipo	Descripción	Constraints
sales_rep_jnid	VARCHAR(100)	JNID del sales rep asignado	INDEXED
sales_rep_name	VARCHAR(255)	Nombre completo del sales rep	
owners	JSONB	Array de propietarios/usuarios asignados	

3.7 Relaciones

Relaciones del estimate con otros registros

Campo	Tipo	Descripción	Constraints
related	JSONB	Array de relaciones con jobs, contacts, etc.	

3.8 Notas y Términos

Campos de texto largo para notas y condiciones

Campo	Tipo	Descripción	Constraints
note	TEXT	Nota visible para el cliente	
internal_note	TEXT	Nota interna (no visible para cliente)	
terms	TEXT	Términos y condiciones del estimate	

3.9 Template y Location

Información de plantilla y ubicación

Campo	Tipo	Descripción	Constraints
template_id	VARCHAR(100)	ID de la plantilla utilizada	
location_id	BIGINT	ID de la ubicación/sucursal	

3.10 Custom Fields y Metadata

Campos personalizados y metadata de control

Campo	Tipo	Descripción	Constraints
custom_fields	JSONB	Campos personalizados definidos por usuario	
is_active	BOOLEAN	Indica si el estimate está activo	DEFAULT TRUE
is_archived	BOOLEAN	Indica si el estimate está archivado	DEFAULT FALSE
created_at	TIMESTAMP	Timestamp de creación en PostgreSQL	DEFAULT CURRENT_TIMESTAMP
updated_at	TIMESTAMP	Timestamp de última actualización en PostgreSQL	DEFAULT CURRENT_TIMESTAMP

4. ÍNDICES Y OPTIMIZACIÓN

La tabla `jobnimbus_estimates` cuenta con **8 índices** estratégicamente diseñados para optimizar las consultas más frecuentes relacionadas con búsqueda, filtrado y agrupamiento de estimates.

Índice	Campo(s)	Tipo	Propósito
<code>idx_jn_estimates_instance</code>	<code>instance</code>	B-Tree	Filtrado por instancia (stamford/guilford)
<code>idx_jn_estimates_jnid</code>	<code>jnid</code>	B-Tree	Búsqueda rápida por JobNimbus ID
<code>idx_jn_estimates_number</code>	<code>number</code>	B-Tree	Búsqueda por número de estimate
<code>idx_jn_estimates_status</code>	<code>status</code>	B-Tree	Filtrado por estado (Draft/Sent/Approved)
<code>idx_jn_estimates_date_created</code>	<code>date_created</code>	B-Tree	Ordenamiento y filtrado por fecha creación
<code>idx_jn_estimates_date_approved</code>	<code>date_approved</code>	B-Tree	Filtrado por fecha de aprobación
<code>idx_jn_estimates_sales_rep</code>	<code>sales_rep_jnid</code>	B-Tree	Filtrado por sales rep
<code>idx_jn_estimates_items</code>	<code>items</code>	GIN	Búsqueda dentro del array de items

4.1 Comandos SQL de Creación de Índices

```
CREATE INDEX idx_jn_estimates_instance ON jobnimbus_estimates(instance);
CREATE INDEX idx_jn_estimates_jnid ON jobnimbus_estimates(jnid);
CREATE INDEX idx_jn_estimates_number ON jobnimbus_estimates(number);
CREATE INDEX idx_jn_estimates_status ON jobnimbus_estimates(status);
CREATE INDEX idx_jn_estimates_date_created ON jobnimbus_estimates(date_created);
CREATE INDEX idx_jn_estimates_date_approved ON jobnimbus_estimates(date_approved);
CREATE INDEX idx_jn_estimates_sales_rep ON jobnimbus_estimates(sales_rep_jnid);
CREATE INDEX idx_jn_estimates_items ON jobnimbus_estimates USING gin(items);
```

5. CAMPOS JSONB FLEXIBLES

Los campos JSONB son fundamentales en la tabla estimates, ya que permiten almacenar estructuras complejas como los items del presupuesto, secciones, propietarios y relaciones de manera flexible y eficiente.

5.1 Campo 'items' - Items del Estimate

El campo **items** almacena un array de objetos que representan los productos o servicios incluidos en el estimate. Cada item contiene información sobre cantidad, precio unitario, costo y descripción.

```
{
  "items": [
    {
      "jnid": "ltor27rgv0jet7njskhoiyg",
      "name": "Gutter Replacement",
      "description": "Full gutter system replacement...",
      "item_type": "material",
      "quantity": 1,
      "uom": "Items",
      "price": 3430.00,
      "cost": 2057.96,
      "amount": 3430.00
    }
  ]
}
```

5.2 Campo 'sections' - Secciones Organizativas

El campo **sections** permite organizar los items en secciones lógicas, útil para estimates complejos con múltiples áreas de trabajo.

```
{
  "sections": [
    {
      "index": 0,
      "group": 0,
      "name": "Roofing Work",
      "showGroupTotal": true
    },
    {
      "index": 1,
      "group": 1,
      "name": "Gutter Installation",
      "showGroupTotal": true
    }
  ]
}
```

```
}
]
```

5.3 Campo 'owners' - Propietarios

El campo **owners** almacena un array de usuarios asignados al estimate.

```
{
  "owners": [
    {
      "id": "mealusrfnkmgola7de3zvn3"
    }
  ]
}
```

5.4 Campo 'related' - Relaciones

El campo **related** vincula el estimate con jobs, contacts y otros registros.

```
{
  "related": [
    {
      "id": "mhnjtcc67sxmgiaw9gswcta",
      "type": "job",
      "number": "1921",
      "name": "422 High Ridge Road"
    },
    {
      "id": "mhnjsps0ghvdx4ni7mcsjo8",
      "type": "contact",
      "number": "2010",
      "name": "Brian Uy"
    }
  ]
}
```

5.5 Ejemplos de Consultas JSONB

```
-- Buscar estimates con un item específico
SELECT jnid, number, total
FROM jobnimbus_estimates
WHERE items @> '[{"name": "Gutter Replacement"}]';
```

```
-- Extraer información de items
SELECT
  jnid,
  number,
  jsonb_array_elements(items)->>'name' as item_name,
  (jsonb_array_elements(items)->>'price')::decimal as price
FROM jobnimbus_estimates
WHERE instance = 'stamford';

-- Buscar estimates relacionados con un job específico
SELECT jnid, number, total
FROM jobnimbus_estimates
WHERE related @> '["type": "job", "id": "job_jnid_123"]';
```

6. EJEMPLOS DE DATOS REALES

A continuación se presentan ejemplos de registros reales de la tabla `jobnimbus_estimates`.

6.1 Ejemplo: Estimate Enviado (Sent)

```
{
  "jnid": "9d55286f-bc73-4ece-ac7a-d4fc7e700b2c",
  "number": "1902",
  "status": 2,
  "status_name": "Sent",
  "date_created": 1763082011,
  "date_estimate": 1763082010,
  "subtotal": 3430.00,
  "tax": 0.00,
  "total": 3430.00,
  "cost": 2057.96,
  "margin": 66.67,
  "sales_rep_name": "Bill Tyson",
  "signed": false,
  "items": [
    {
      "name": "Unnamed Product",
      "quantity": 1,
      "price": 3430.00,
      "cost": 2057.96,
      "amount": 3430.00
    }
  ]
}
```

6.2 Ejemplo: Estimate Aprobado (Approved)

```
{
  "jnid": "b7345d0a-d9d9-4742-b395-7d070aa6e63e",
  "number": "1900",
  "status": 4,
  "status_name": "Approved",
  "date_created": 1762902594,
  "date_approved": 1763058311,
  "subtotal": 6000.00,
  "tax": 0.00,
  "total": 6000.00,
  "cost": 3599.93,
  "margin": 66.67,
  "sales_rep_name": "Bill Tyson",
}
```

```
"signature_status": "Fully Signed",
"items": [
  {
    "name": "Harvey Windows",
    "quantity": 1,
    "price": 6000.00,
    "cost": 3599.93
  }
]
}
```

7. RELACIONES CON OTRAS TABLAS

Los estimates tienen relaciones clave con jobs, contacts e invoices.

Tabla Relacionada	Campo de Unión	Tipo	Descripción
jobnimbus_jobs	related JSONB	1:N	Un estimate puede relacionarse con un job
jobnimbus_contacts	related JSONB	1:N	Un estimate tiene un contacto asociado
jobnimbus_invoices	estimate_jnid	1:1	Un estimate puede convertirse en invoice
jobnimbus_products	items->jnid	N:N	Items referencian productos del catálogo

8. EJEMPLOS DE QUERIES COMUNES

8.1 Listar Estimates Pendientes de Aprobación

```
SELECT number, total, sales_rep_name,  
       TO_TIMESTAMP(date_sent) as sent_date  
FROM jobnimbus_estimates  
WHERE instance = 'stamford'  
      AND status = 2  
      AND is_active = TRUE  
ORDER BY date_sent DESC;
```

8.2 Calcular Total de Estimates Aprobados por Mes

```
SELECT  
  DATE_TRUNC('month', TO_TIMESTAMP(date_approved)) as month,  
  COUNT(*) as count,  
  SUM(total) as total_approved  
FROM jobnimbus_estimates  
WHERE instance = 'stamford'  
      AND status = 4  
      AND date_approved IS NOT NULL  
GROUP BY month  
ORDER BY month DESC;
```

8.3 Estimates por Sales Rep

```
SELECT  
  sales_rep_name,  
  COUNT(*) as total_estimates,  
  SUM(CASE WHEN status = 4 THEN 1 ELSE 0 END) as approved,  
  SUM(CASE WHEN status = 4 THEN total ELSE 0 END) as total_value  
FROM jobnimbus_estimates  
WHERE instance = 'stamford'  
      AND is_active = TRUE  
GROUP BY sales_rep_name  
ORDER BY total_value DESC;
```

8.4 Análisis de Márgenes

```
SELECT  
  number,  
  total,  
  cost,
```

```
margin,  
  (total - cost) as profit  
FROM jobnimbus_estimates  
WHERE instance = 'stamford'  
  AND status = 4  
  AND margin IS NOT NULL  
ORDER BY profit DESC  
LIMIT 10;
```

9. MEJORES PRÁCTICAS

9.1 Validación de Datos

- Validar que subtotal + tax = total antes de guardar
- Verificar que margin se calcule correctamente: $((\text{total} - \text{cost}) / \text{total}) * 100$
- Asegurar que items sea un array válido con estructura correcta

9.2 Uso de Índices

- Usar idx_jn_estimates_status para filtrar por estado
- Aprovechar idx_jn_estimates_sales_rep para reportes de ventas
- Usar idx_jn_estimates_items para búsquedas dentro de items

9.3 Manejo de Estados

- Siempre actualizar date_updated al cambiar estado
- Registrar date_approved cuando status cambie a 4
- No eliminar estimates físicamente, usar is_active = FALSE

9.4 Cálculos Financieros

- Recalcular total cuando se modifiquen items
- Mantener consistencia entre items individuales y totales
- Documentar cualquier descuento en custom_fields

10. APÉNDICE

10.1 Códigos de Estado

Código	Nombre	Descripción
1	Draft	Borrador, no enviado al cliente
2	Sent	Enviado al cliente, pendiente de respuesta
4	Approved	Aprobado por el cliente
5	Rejected	Rechazado por el cliente