

You already have isEditMode, so you just need to **make the header styles conditional**.

Change ONLY the DialogTitle sx background (and optionally border) based on isEditMode.

 Replace your DialogTitle sx with this:

```
<DialogTitle  
sx={{  
  fontWeight: 800,  
  display: "flex",  
  alignItems: "center",  
  justifyContent: "space-between",  
  pr: 2,  
  background: isEditMode  
    ? "linear-gradient(135deg, #f59e0b 0%, #ea580c 100%)" // ORANGE (Edit)  
    : "linear-gradient(135deg, #1e3a5f 0%, #2d5a8c 100%)" // BLUE (View)  
  color: "#ffffff",  
  borderBottom: isEditMode  
    ? "3px solid #fb923c"  
    : "3px solid #60a5fa",  
  py: 2.5,  
  transition: "all 0.3s ease", // smooth color change  
}}>
```

Result

- **View Mode** → Professional **Blue header**
- **Edit Mode** → Professional **Orange header**
- Smooth animated transition
- No logic changes needed — purely styling

Optional polish (recommended)

You can also make the icon change color slightly in edit mode:

```
<Box sx={{ fontSize: 28 }}>  
{isEditMode ? "📝" : "📋"}  
</Box>
```

If you want:

- Gradient animation
- Pulsing edit header
- Different orange shades (amber / warning / deep orange)

Just tell me 

Budgetary

Below is the **updated and regenerated version** of your dialog code, modified **only to align with the new field names you provided**.

What I changed (mapping old → new)

Old Field	New Field
tenderName	bqTitle
tenderReferenceNo	referenceNo
customerName	customerName
customerAddress	customerAddress
leadOwner	leadOwner
civilOrDefence	defenceAndNonDefence
estimatedValueInCrWithoutGST	estimateValueInCrWithoutGST
submittedValueInCrWithoutGST	submittedValueInCrWithoutGST
lastDateOfSub	dateOfLetterSubmission
competitorsInfo	JSON_competitors
presentStatus	presentStatus

Unused / removed fields were **not reintroduced**.

◆ UPDATED DIALOG CODE (Clean & Consistent)

You can **copy-paste directly**. Styling and structure remain unchanged.

```
{/* EDIT DIALOG */}

<Dialog

  open={editDialogOpen}

  onClose={handleEditCancel}

  maxWidth="xl"
```

```
fullWidth
>
{/* HEADER */}

<DialogTitle sx={{ background: "linear-gradient(135deg, #1e3a5f, #2d5a8c)", color: "#fff" }}>

  <Box display="flex" justifyContent="space-between" alignItems="center">

    <Box>

      <Typography fontWeight={800}>
        {editingRow?.bqTitle || "Lead Details"}
      </Typography>

      <Typography variant="caption">
        Reference: {editingRow?.referenceNo || "N/A"}
      </Typography>

    </Box>

    <IconButton onClick={handleEditCancel} sx={{ color: "#fff" }}>
      <CloseRounded />
    </IconButton>

  </Box>

</DialogTitle>

<DialogContent sx={{ background: "#f8fafc" }}>

  {/* BASIC DETAILS */}

  <Box sx={{ mb: 3 }}>
    <Typography fontWeight={800}>Tender Information</Typography>
  </Box>

  <Box
    sx={{

      display: "grid",
```

```

gridTemplateColumns: "repeat(auto-fit, minmax(220px, 1fr))",
gap: 1.5,
}}
>
{[
 { label: "BQ Title", key: "bqTitle" },
 { label: "Reference No", key: "referenceNo" },
 { label: "Date of Letter Submission", key: "dateOfLetterSubmission", isDate: true },
].map((field) => (
<Box key={field.key} sx={{ p: 2, border: "1px solid #e0e7ff", borderRadius: 2 }}>
<Typography variant="caption">{field.label}</Typography>
{isEditMode ? (
<TextField
  type={field.isDate ? "date" : "text"}
  value={editingRow?.[field.key] || ""}
  onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
  fullWidth
  size="small"
  InputLabelProps={field.isDate ? { shrink: true } : undefined}
/>
) : (
<Typography>{editingRow?.[field.key] || "-"}</Typography>
)}
</Box>
))}

</Box>
</Box>

```

```

/* CUSTOMER INFORMATION */

<Box sx={{ mb: 3 }}>
  <Typography fontWeight={800}>Customer Information</Typography>

  <Box sx={{ display: "grid", gridTemplateColumns: "repeat(auto-fit, minmax(220px, 1fr))", gap: 1.5 }}>
    {[

      { label: "Customer Name", key: "customerName" },
      { label: "Lead Owner", key: "leadOwner" },
      { label: "Defence / Non-Defence", key: "defenceAndNonDefence" },
    ].map((field) => (
      <Box key={field.key} sx={{ p: 2, border: "1px solid #e0e7ff", borderRadius: 2 }}>
        <Typography variant="caption">{field.label}</Typography>
        {isEditMode ? (
          <TextField
            value={editingRow?.[field.key] || ""}
            onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
            fullWidth
            size="small"
          />
        ) : (
          <Typography>{editingRow?.[field.key] || "-"}</Typography>
        )}
      </Box>
    )));
  </Box>

```

/* ADDRESS */

```

<Box sx={{ mt: 2, p: 2, border: "1px solid #e0e7ff", borderRadius: 2 }}>
  <Typography variant="caption">Customer Address</Typography>
  {isEditMode ? (
    <TextField
      value={editingRow?.customerAddress || ""}
      onChange={(e) => handleEditFieldChange("customerAddress", e.target.value)}
      fullWidth
      multiline
      minRows={2}
    />
  ) : (
    <Typography whiteSpace="pre-wrap">{editingRow?.customerAddress || "-"}</Typography>
  )}
</Box>
</Box>

/* FINANCIAL DETAILS */

<Box sx={{ mb: 3 }}>
  <Typography fontWeight={800}>Financial Details</Typography>

  <Box sx={{ display: "grid", gridTemplateColumns: "repeat(auto-fit, minmax(220px, 1fr))", gap: 1.5 }}>
    {[{
      { label: "Estimated Value (Cr, w/o GST)", key: "estimateValueInCrWithoutGST" },
      { label: "Submitted Value (Cr, w/o GST)", key: "submittedValueInCrWithoutGST" },
    ].map((field) => (
      <Box key={field.key} sx={{ p: 2, border: "1px solid #e0e7ff", borderRadius: 2 }}>
        <Typography variant="caption">{field.label}</Typography>
    ))
  }
</Box>

```

```

{isEditMode ? (
  <TextField
    value={editingRow?.[field.key] || ""}
    onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
    fullWidth
    size="small"
  />
) : (
  <Typography>{editingRow?.[field.key] || "-"}</Typography>
)
)}
```

</Box>

```

))}

</Box>

</Box>
```

{/* STATUS & COMPETITORS */}

```

<Box sx={{ mb: 2 }}>
  <Typography fontWeight={800}>Status & Competitors</Typography>

```

<Box sx={{ display: "grid", gridTemplateColumns: "repeat(auto-fit, minmax(220px, 1fr))", gap: 1.5 }}>

```

{[
  { label: "Present Status", key: "presentStatus" },
  { label: "Competitors (JSON)", key: "JSON_competitors" },
].map((field) => (
  <Box key={field.key} sx={{ p: 2, border: "1px solid #e0e7ff", borderRadius: 2 }}>
    <Typography variant="caption">{field.label}</Typography>
    {isEditMode ? (

```

```
<TextField
  value={editingRow?.[field.key] || ""}
  onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
  fullWidth
  multiline
  minRows={2}
/>

): (
  <Typography whiteSpace="pre-wrap">{editingRow?.[field.key] || "-"
}</Typography>
)}
```

})

```
</Box>
))}

</Box>
</Box>
</DialogContent>

 {/* ACTIONS */}

<DialogActions>
  {!isEditMode ? (
    <Button onClick={handleEnterEditMode} variant="contained">
      Edit
    </Button>
  ): (
    <>
      <Button onClick={handleCancelEdit}>Cancel</Button>
      <Button onClick={handleEditSave} variant="contained" color="success">
        Save
      </Button>
    </>
  )
}
```

```
</Button>  
</>  
}  
</DialogActions>  
</Dialog>
```

If you want, I can also:

- Convert JSON_competitors into a **dynamic table**
- Add **dropdowns** for presentStatus / defenceAndNonDefence
- Add **numeric validation** for value fields
- Normalize this to a **field-config-driven reusable dialog**

Just tell me 

Lead Submitted

Got it! You want your dialog fields updated to match this new list of keys. I can regenerate the **edit/view dialog** code accordingly, keeping your professional blue theme and layout intact.

Here's a mapping of your new fields for clarity:

Section	Fields
Tender Info	tenderName, tenderReferenceNo, tenderType, website, tenderDate
Customer/Bid Info	customerName, customerAddress, bidOwner, rfpReceivedOn
Financial/Approval	valueEMDInCrore, rfpDueDate, dmktgInPrincipalApprovalRxdOn, sellingPriceApprovalInitiatedOn, bidSubmittedOn,
Dates	approvalSBUFinanceOn, approvalGMOOn, sentToFinanceGMDmktgApprovalRxdOn, dmktgApprovalRxdOn
Status	presentStatus

Here's the **updated regenerated Dialog code** for all these fields:

```
{/* EDIT DIALOG */}

<Dialog

  open={editDialogOpen}

  onClose={handleEditCancel}

  maxWidth="xl"

  fullWidth

  PaperProps={{

    sx: {

      borderRadius: 3,

      overflow: "hidden",

      background: "#ffffff",

      boxShadow: "0 25px 50px rgba(0,0,0,0.15), 0 10px 30px rgba(30,64,95,0.2)",

    }

  }

}
```

```
        maxHeight: "90vh",
    },
})

BackdropProps={

sx: {

    backdropFilter: "blur(5px)",

    backgroundColor: "rgba(0, 0, 0, 0.4)",

},
}

>

{/* HEADER */}

<DialogTitle

sx={{

fontWeight: 800,

display: "flex",

alignItems: "center",

justifyContent: "space-between",

pr: 2,

background: "linear-gradient(135deg, #1e3a5f 0%, #2d5a8c 100%)",

color: "#ffffff",

borderBottom: "3px solid #60a5fa",

py: 2.5,
}}>

>

<Box display="flex" alignItems="center" gap={2}>

<Box sx={{ fontSize: 28, fontWeight: 800 }}>  </Box>

<Box>

<Typography variant="h6" sx={{ fontWeight: 800, color: "#ffffff" }}>
```

```
{editingRow?.tenderName || "Lead Details"}  
</Typography>  
<Typography variant="caption" sx={{ color: "#bfdbfe", mt: 0.5 }}>  
  Reference: {editingRow?.tenderReferenceNo || "N/A"}  
</Typography>  
</Box>  
</Box>  
<Box display="flex" alignItems="center" gap={2}>  
  <Chip  
    label={isEditMode ? "EDIT MODE" : "VIEW MODE"}  
    size="small"  
    sx={{  
      fontWeight: 700,  
      fontSize: "0.75rem",  
      background: isEditMode ? "#fbbf24" : "#60a5fa",  
      color: isEditMode ? "#1f2937" : "#ffffff",  
    }}  
  />  
  <IconButton  
    onClick={handleEditCancel}  
    sx={{ color: "#ffffff", "&:hover": { backgroundColor: "rgba(255,255,255,0.1)" } }}>  
    >  
    <CloseRounded />  
  </IconButton>  
</Box>  
</DialogTitle>  
  
<DialogContent
```

```
sx={{  
    background: "#f8fafc",  
    p: 2.5,  
    maxHeight: "calc(90vh - 130px)",  
    overflowY: "auto",  
    overflowX: "hidden",  
}}  
>  
/* TENDER INFORMATION */  
<Box sx={{ mb: 3 }}>  
  <Typography variant="subtitle1" sx={{ fontWeight: 800, color: "#1e3a5f", mb: 2 }}>  
    Tender Information  
  </Typography>  
<Box  
  sx={{  
    display: "grid",  
    gridTemplateColumns: { xs: "1fr", sm: "repeat(2,1fr)", md: "repeat(3,1fr)", lg: "repeat(4,1fr)" },  
    gap: 1.5,  
}}>  
  {[  
    { label: "Tender Name", key: "tenderName" },  
    { label: "Reference No", key: "tenderReferenceNo" },  
    { label: "Tender Type", key: "tenderType" },  
    { label: "Website", key: "website" },  
    { label: "Tender Date", key: "tenderDate", isDate: true },  
  ].map((field) => (
```

```

<Box key={field.key} sx={{ background: "#fff", border: "1px solid #e0e7ff",
borderRadius: 2, p: 2 }}>

  <Typography variant="caption" sx={{ color: "#64748b"
}}>{field.label}</Typography>

  {isEditMode ? (
    <TextField
      type={field.isDate ? "date" : "text"}
      value={editingRow?.[field.key] || ""}
      onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
      fullWidth
      size="small"
      InputLabelProps={field.isDate ? { shrink: true } : undefined}
      sx={{ mt: 1 }}
    />
  ) : (
    <Typography sx={{ mt: 1 }}>{editingRow?.[field.key] || "-"}</Typography>
  )}
</Box>
))}

</Box>
</Box>

```

```

{/* CUSTOMER / BID INFORMATION */}

<Box sx={{ mb: 3 }}>

  <Typography variant="subtitle1" sx={{ fontWeight: 800, color: "#1e3a5f", mb: 2 }}>
    Customer / Bid Information
  </Typography>

  <Box
    sx={{
```

```

        display: "grid",
        gridTemplateColumns: { xs: "1fr", sm: "repeat(2,1fr)", md: "repeat(3,1fr)", lg: "repeat(4,1fr)" },
        gap: 1.5,
    }}
>
{[
    { label: "Customer Name", key: "customerName" },
    { label: "Bid Owner", key: "bidOwner" },
    { label: "RFP Received On", key: "rfpReceivedOn", isDate: true },
].map((field) => (
    <Box key={field.key} sx={{ background: "#fff", border: "1px solid #e0e7ff", borderRadius: 2, p: 2 }}>
        <Typography variant="caption" sx={{ color: "#64748b" }}>{field.label}</Typography>
    {isEditMode ? (
        <TextField
            type={field.isDate ? "date" : "text"}
            value={editingRow?.[field.key] || ""}
            onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
            fullWidth
            size="small"
            InputLabelProps={field.isDate ? { shrink: true } : undefined}
            sx={{ mt: 1 }}
        />
    ) : (
        <Typography sx={{ mt: 1 }}>{editingRow?.[field.key] || "-"}</Typography>
    )}
</Box>

```

```

        ))}
      </Box>

      <Box sx={{ mt: 2, background: "#fff", border: "1px solid #e0e7ff", borderRadius: 2, p: 2 }}>
        <Typography variant="caption" sx={{ color: "#64748b" }}>Customer Address</Typography>
        {isEditMode ? (
          <TextField
            value={editingRow?.customerAddress || ""}
            onChange={(e) => handleEditFieldChange("customerAddress", e.target.value)}
            fullWidth
            multiline
            minRows={2}
            sx={{ mt: 1 }}
          />
        ) : (
          <Typography sx={{ mt: 1, whiteSpace: "pre-wrap" }}>{editingRow?.customerAddress || "-"}</Typography>
        )}
      </Box>
    </Box>

    /* FINANCIAL / APPROVAL DATES */

    <Box sx={{ mb: 3 }}>
      <Typography variant="subtitle1" sx={{ fontWeight: 800, color: "#1e3a5f", mb: 2 }}>
        Financial & Approval Dates
      </Typography>
      <Box

```

```

sx={{{
    display: "grid",
    gridTemplateColumns: { xs: "1fr", sm: "repeat(2,1fr)", md: "repeat(3,1fr)", lg: "repeat(4,1fr)" },
    gap: 1.5,
  }}
>
{[
  { label: "Value of EMD (Cr)", key: "valueEMDInCrore" },
  { label: "RFP Due Date", key: "rfpDueDate", isDate: true },
  { label: "DMktg Principal Approval Rx'd On", key: "dmktgInPrincipalApprovalRxdOn", isDate: true },
  { label: "Selling Price Approval Initiated On", key: "sellingPriceApprovalInitiatedOn", isDate: true },
  { label: "Bid Submitted On", key: "bidSubmittedOn", isDate: true },
  { label: "Approval SBU Finance On", key: "approvalSBUFinanceOn", isDate: true },
  { label: "Approval GM On", key: "approvalGMOOn", isDate: true },
  { label: "Sent to Finance GM / DMktg Approval Rx'd On", key: "sentToFinanceGMDmktgApprovalRxdOn", isDate: true },
  { label: "DMktg Approval Rx'd On", key: "dmktgApprovalRxdOn", isDate: true },
].map((field) => (
  <Box key={field.key} sx={{ background: "#fff", border: "1px solid #e0e7ff", borderRadius: 2, p: 2 }}>
    <Typography variant="caption" sx={{ color: "#64748b" }}>{field.label}</Typography>
    {isEditMode ? (
      <TextField
        type={field.isDate ? "date" : "text"}
        value={editingRow?.[field.key] || ""}
        onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
      >
    ) : (
      <Text>{field.value}</Text>
    )}
  </Box>
))
}

```

```

    fullWidth
    size="small"
    InputLabelProps={field.isDate ? { shrink: true } : undefined}
    sx={{ mt: 1 }}
  />
): (
<Typography sx={{ mt: 1 }}>{editingRow?.[field.key] || "-"}</Typography>
)
</Box>
))}

</Box>
</Box>

/* STATUS */
<Box sx={{ mb: 2 }}>
<Typography variant="subtitle1" sx={{ fontWeight: 800, color: "#1e3a5f", mb: 2 }}>
  Status
</Typography>
<Box sx={{ display: "grid", gridTemplateColumns: { xs: "1fr", sm: "1fr", md: "1fr" }, gap: 1.5 }}>
  <Box sx={{ background: "#fff", border: "1px solid #e0e7ff", borderRadius: 2, p: 2 }}>
    <Typography variant="caption" sx={{ color: "#64748b" }}>Present
    Status</Typography>
    {isEditMode ? (
      <TextField
        value={editingRow?.presentStatus || ""}
        onChange={(e) => handleEditFieldChange("presentStatus", e.target.value)}
        fullWidth
        size="small"
      )
    ) : (
      <Text>{editingRow?.presentStatus}</Text>
    )}
  </Box>
</Box>

```

```
    sx={{ mt: 1 }}

  />

  ) : (
    <Typography sx={{ mt: 1 }}>{editingRow?.presentStatus || "-"}</Typography>
  )}

</Box>

</Box>

</Box>

</DialogContent>

{/* DIALOG ACTIONS */}

<DialogActions sx={{ background: "#f8fafc", borderTop: "1px solid #e0e7ff", p: 2 }}>

  {!isEditMode ? (
    <Button onClick={handleEnterEditMode} variant="contained" sx={{ background: "#1e3a5f", color: "#fff" }}>
      Edit Details
    </Button>
  ) : (
    <>
      <Button onClick={handleCancelEdit}>Cancel</Button>
      <Button onClick={handleEditSave} variant="contained" sx={{ background: "#10b981", color: "#fff" }}>
        Save Changes
      </Button>
    </>
  )}
</DialogActions>

</Dialog>
```

If you want, I can also **group these date fields into a timeline-style visual** like your original design—it will make approvals and submissions much easier to scan visually.

Do you want me to do that next?

Domestic Leads

Perfect! You want the dialog regenerated for **this specific set of fields**, keeping the **professional blue theme**, with **edit/view modes**, grouped into sections like Tender Info, Customer Info, Financial Details, Timeline, Status & Results, and Additional Info.

Here's a **clean, updated version** for your new field structure:

```
{/* EDIT DIALOG */}

<Dialog
  open={editDialogOpen}
  onClose={handleEditCancel}
  maxWidth="xl"
  fullWidth
  PaperProps={{
    sx: {
      borderRadius: 3,
      overflow: "hidden",
      background: "#ffffff",
      boxShadow: "0 25px 50px rgba(0,0,0,0.15), 0 10px 30px rgba(30,64,95,0.2)",
      maxHeight: "90vh",
    },
  }}
  BackdropProps={}
```

```
sx: { backdropFilter: "blur(5px)", backgroundColor: "rgba(0,0,0,0.4)" },  
}  
>  
{/* HEADER */}  
<DialogTitle  
sx={{  
fontWeight: 800,  
display: "flex",  
alignItems: "center",  
justifyContent: "space-between",  
pr: 2,  
background: "linear-gradient(135deg, #1e3a5f 0%, #2d5a8c 100%)",  
color: "#ffffff",  
borderBottom: "3px solid #60a5fa",  
py: 2.5,  
}}  
>  
<Box display="flex" alignItems="center" gap={2}>  
  <Box sx={{ fontSize: 28, fontWeight: 800 }}>  </Box>  
  <Box>  
    <Typography variant="h6" sx={{ fontWeight: 800, color: "#ffffff" }}>  
      {editingRow?.tenderName || "Lead Details"}  
    </Typography>  
    <Typography variant="caption" sx={{ color: "#bfdbfe", mt: 0.5 }}>  
      Reference: {editingRow?.tenderReferenceNo || "N/A"}  
    </Typography>  
  </Box>  
</Box>
```

```
<Box display="flex" alignItems="center" gap={2}>
  <Chip
    label={isEditMode ? "EDIT MODE" : "VIEW MODE"}
    size="small"
    sx={{
      fontWeight: 700,
      fontSize: "0.75rem",
      background: isEditMode ? "#fbbf24" : "#60a5fa",
      color: isEditMode ? "#1f2937" : "#ffffff",
    }}
  />
  <IconButton
    onClick={handleEditCancel}
    sx={{ color: "#ffffff", "&:hover": { backgroundColor: "rgba(255,255,255,0.1)" } }}
  >
    <CloseRounded />
  </IconButton>
</Box>
</DialogTitle>

<DialogContent
  sx={{
    background: "#f8fafc",
    p: 2.5,
    maxHeight: "calc(90vh - 130px)",
    overflowY: "auto",
    overflowX: "hidden",
  }}>
```

```

    }}

>

/* TENDER INFORMATION */

<Box sx={{ mb: 3 }}>

<Typography variant="subtitle1" sx={{ fontWeight: 800, color: "#1e3a5f", mb: 2 }}>
  Tender Information
</Typography>

<Box
  sx={{

    display: "grid",

    gridTemplateColumns: { xs: "1fr", sm: "repeat(2,1fr)", md: "repeat(2,1fr)", lg:
      "repeat(4,1fr)" },

    gap: 1.5,
  }}
>

{[

  { label: "Tender Name", key: "tenderName" },

  { label: "Tender Reference No", key: "tenderReferenceNo" },

  { label: "Tender Type", key: "tenderType" },

  { label: "Document Type", key: "documentType" },

  { label: "Tender Dated", key: "tenderDated", isDate: true },
]

.map((field) => (
  <Box key={field.key} sx={{ background: "#fff", border: "1px solid #e0e7ff",
  borderRadius: 2, p: 2 }}>
    <Typography variant="caption" sx={{ color: "#64748b"
  }}>{field.label}</Typography>
    {isEditMode ? (
      <TextField
        type={field.isDate ? "date" : "text"}>
    ) : (
      <Text>{field.value}</Text>
    )}
  </Box>
))}


```

```

        value={editingRow?.[field.key] || ""}
        onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
        fullWidth
        size="small"
        InputLabelProps={field.isDate ? { shrink: true } : undefined}
        sx={{ mt: 1 }}
      />
    ) : (
      <Typography sx={{ mt: 1 }}>{editingRow?.[field.key] || "-"}</Typography>
    )}
```

</Box>

)}}

</Box>

</Box>

```

/* CUSTOMER INFORMATION */

<Box sx={{ mb: 3 }}>

  <Typography variant="subtitle1" sx={{ fontWeight: 800, color: "#1e3a5f", mb: 2 }}>
    Customer Information
  </Typography>

  <Box
    sx={{
      display: "grid",
      gridTemplateColumns: { xs: "1fr", sm: "repeat(2,1fr)", md: "repeat(2,1fr)", lg: "repeat(4,1fr)" },
      gap: 1.5,
    }}
  >
```

```

{[
  { label: "Customer Name", key: "customerName" },
  { label: "Lead Owner", key: "leadOwner" },
  { label: "Civil / Defence", key: "civilOrDefence" },
  { label: "Business Domain", key: "businessDomain" },
].map((field) => (
  <Box key={field.key} sx={{ background: "#fff", border: "1px solid #e0e7ff",
borderRadius: 2, p: 2 }}>
    <Typography variant="caption" sx={{ color: "#64748b"
}}>{field.label}</Typography>
    {isEditMode ? (
      <TextField
        value={editingRow?.[field.key] || ""}
        onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
        fullWidth
        size="small"
        sx={{ mt: 1 }}
      />
    ) : (
      <Typography sx={{ mt: 1 }}>{editingRow?.[field.key] || "-"}</Typography>
    )}
  </Box>
))}

</Box>

/* Customer Address - Full Width */

<Box sx={{ mt: 2, background: "#fff", border: "1px solid #e0e7ff", borderRadius: 2, p: 2
}}>

```

```
<Typography variant="caption" sx={{ color: "#64748b" }}>Customer
Address</Typography>

{isEditMode ? (
  <TextField
    value={editingRow?.customerAddress || ""}
    onChange={(e) => handleEditFieldChange("customerAddress", e.target.value)}
    fullWidth
    size="small"
    multiline
    minRows={2}
    sx={{ mt: 1 }}
  />
):(
  <Typography sx={{ mt: 1, whiteSpace: "pre-wrap" }}>{editingRow?.customerAddress
  || "-"}</Typography>
)}
```

</Box>

</Box>

{/* FINANCIAL DETAILS */}

```
<Box sx={{ mb: 3 }}>
  <Typography variant="subtitle1" sx={{ fontWeight: 800, color: "#1e3a5f", mb: 2 }}>
    Financial Details
  </Typography>
  <Box
    sx={{
      display: "grid",
      gridTemplateColumns: { xs: "1fr", sm: "repeat(2,1fr)", md: "repeat(2,1fr)", lg:
        "repeat(4,1fr)" },
    }}>
```

```

    gap: 1.5,
  }
>
{[
  { label: "Value of EMD", key: "valueOfEMD" },
  { label: "Estimated Value (Cr, w/o GST)", key: "estimatedValueInCrWithoutGST" },
  { label: "Submitted Value (Cr, w/o GST)", key: "submittedValueInCrWithoutGST" },
  { label: "Order Won Value (Cr, w/o GST)", key: "orderWonValueInCrWithoutGST" },
].map((field) => (
  <Box key={field.key} sx={{ background: "#fff", border: "1px solid #e0e7ff",
borderRadius: 2, p: 2 }}>
    <Typography variant="caption" sx={{ color: "#64748b"
}}>{field.label}</Typography>
    {isEditMode ? (
      <TextField
        value={editingRow?.[field.key] || ""}
        onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
        fullWidth
        size="small"
        sx={{ mt: 1 }}
      />
    ) : (
      <Typography sx={{ mt: 1 }}>{editingRow?.[field.key] || "-"}</Typography>
    )}
  </Box>
))}

</Box>

</Box>

```

```
/* TIMELINE */\n\n<Box sx={{ mb: 3 }}>\n  <Typography variant="subtitle1" sx={{ fontWeight: 800, color: "#1e3a5f", mb: 2 }}>\n    Timeline\n  </Typography>\n<Box\n  sx={{\n    display: "grid",\n    gridTemplateColumns: { xs: "1fr", sm: "repeat(2,1fr)", md: "repeat(2,1fr)", lg: "repeat(3,1fr)" },\n    gap: 1.5,\n  }}\n>\n{[\n  {\n    label: "Last Date of Submission", key: "lastDateOfSub", isDate: true },\n    label: "Sole / Consortium", key: "soleOrConsortium" },\n    label: "Pre-Bid Meeting Date & Time", key: "prebidMeetingDateTime" },\n].map((field) => (\n  <Box key={field.key} sx={{ background: "#fff", border: "1px solid #e0e7ff",\nborderRadius: 2, p: 2 }}>\n    <Typography variant="caption" sx={{ color: "#64748b"\n}}>{field.label}</Typography>\n    {isEditMode ? (\n      <TextField\n        type={field.isDate ? "datetime-local" : "text"}\n        value={editingRow?.[field.key] || ""}\n        onChange={(e) => handleEditFieldChange(field.key, e.target.value)}\n        fullWidth\n        size="small"\n      </TextField>\n    ) : (\n      <Text\n        style={{\n          color: "#1e3a5f",\n          fontWeight: 800,\n        }}\n        onClick={handleEditFieldChange}\n      </Text>\n    )}\n  </Box>\n)}
```

```

InputLabelProps={field.isDate ? { shrink: true } : undefined}
sx={{ mt: 1 }}
/>
): (
<Typography sx={{ mt: 1 }}>{editingRow?.[field.key] || "-"}</Typography>
)}
</Box>
))}

</Box>

</Box>

/* STATUS & RESULTS */

<Box sx={{ mb: 3 }}>
<Typography variant="subtitle1" sx={{ fontWeight: 800, color: "#1e3a5f", mb: 2 }}>
Status & Results
</Typography>
<Box
sx={{
display: "grid",
gridTemplateColumns: { xs: "1fr", sm: "repeat(2,1fr)", md: "repeat(2,1fr)", lg: "repeat(3,1fr)" },
gap: 1.5,
}}
>
{[
{ label: "Won / Lost / Participated", key: "wonLostParticipated" },
{ label: "Open / Closed", key: "openClosed" },
{ label: "Present Status", key: "presentStatus" },

```

```

].map((field) => (
  <Box key={field.key} sx={{ background: "#fff", border: "1px solid #e0e7ff",
borderRadius: 2, p: 2 }}>
    <Typography variant="caption" sx={{ color: "#64748b"
}}>{field.label}</Typography>
    {isEditMode ? (
      <TextField
        value={editingRow?.[field.key] || ""}
        onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
        fullWidth
        size="small"
        sx={{ mt: 1 }}
      />
    ) : (
      <Typography sx={{ mt: 1 }}>{editingRow?.[field.key] || "-"}</Typography>
    )}
  </Box>
))}

</Box>
</Box>

```

```

/* ADDITIONAL INFORMATION */

<Box sx={{ mb: 2 }}>
  <Typography variant="subtitle1" sx={{ fontWeight: 800, color: "#1e3a5f", mb: 2 }}>
    Additional Information
  </Typography>
  <Box
    sx={{{
      display: "grid",

```

```

gridTemplateColumns: { xs: "1fr", sm: "repeat(2,1fr)", md: "repeat(2,1fr)", lg: "repeat(3,1fr)" },
gap: 1.5,
})
>
{[
{ label: "Competitors Info", key: "competitorsInfo" },
{ label: "Reason for Losing/Participating", key: "reasonForLossingOpp" },
{ label: "Corrigendums Date / File", key: "corrigendumsDateFile" },
].map((field) => (
<Box key={field.key} sx={{ background: "#fff", border: "1px solid #e0e7ff",
borderRadius: 2, p: 2 }}>
<Typography variant="caption" sx={{ color: "#64748b"
}}>{field.label}</Typography>
{isEditMode ? (
<TextField
value={editingRow?.[field.key] || ""}
onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
fullWidth
size="small"
multiline
minRows={2}
sx={{ mt: 1 }}
/>
):((
<Typography sx={{ mt: 1, whiteSpace: "pre-wrap" }}>{editingRow?.[field.key] || "-"
}</Typography>
)})
</Box>

```

```

        ))}
      </Box>
    </Box>
  </DialogContent>

  {/* ACTIONS */}

  <DialogActions sx={{ background: "#f8fafc", borderTop: "1px solid #e0e7ff", p: 2 }}>
    {!isEditMode ? (
      <Button onClick={handleEnterEditMode} variant="contained" sx={{ background: "#1e3a5f", color: "#fff" }}>
         Edit Details
      </Button>
    ) : (
      <>
        <Button onClick={handleCancelEdit}>Cancel</Button>
        <Button onClick={handleEditSave} variant="contained" sx={{ background: "#10b981", color: "#fff" }}>
           Save Changes
        </Button>
      </>
    )}
  </DialogActions>
</Dialog>

```

This covers **all your fields exactly as listed**, preserving the **sectional layout, hover effects**, and **professional blue theme**.

I can also create a **field-config-driven version** next, so any future updates to fields only require updating an array rather than editing JSX manually.

Do you want me to do that?

CRM Leads

Absolutely! I've updated your Dialog component code to reflect the new fields you listed: leadID, issueDate, tenderName, organisation, documentType, tenderType, emdInCrore, approxTenderValueCrore, lastDateSubmission, preBidDate, teamAssigned, remarks, and corrigendumInfo. I've mapped them into appropriate sections (Tender Info, Financial, Timeline, and Additional Info) and kept the professional blue theme intact with editable/view modes.

Here's the fully updated code:

```
/* EDIT DIALOG - VIEW MODE & EDIT MODE - PROFESSIONAL BLUE THEME */  
  
<Dialog  
  open={editDialogOpen}  
  onClose={handleEditCancel}  
  maxWidth="xl"  
  fullWidth  
  PaperProps={{  
    sx: {  
      borderRadius: 3,  
      overflow: "hidden",  
      background: "#ffffff",  
      boxShadow: "0 25px 50px rgba(0,0,0,0.15), 0 10px 30px rgba(30,64,95,0.2)",  
      maxHeight: "90vh",  
    },  
  }}  
  BackdropProps={{  
    sx: {  
      backdropFilter: "blur(5px)",  
      backgroundColor: "rgba(0, 0, 0, 0.4)",  
    },  
  },
```

```
  }}

>

{/* HEADER */}

<DialogTitle

sx={{

  fontWeight: 800,

  display: "flex",

  alignItems: "center",

  justifyContent: "space-between",

  pr: 2,

  background: "linear-gradient(135deg, #1e3a5f 0%, #2d5a8c 100%)",

  color: "#ffffff",

  borderBottom: "3px solid #60a5fa",

  py: 2.5,

}

}>

>

<Box display="flex" alignItems="center" gap={2}>

<Box sx={{ fontSize: 28, fontWeight: 800 }}>  </Box>

<Box>

<Typography variant="h6" sx={{ fontWeight: 800, color: "#ffffff" }}>

{editingRow?.tenderName || "Lead Details"}

</Typography>

<Typography variant="caption" sx={{ color: "#bfdbfe", mt: 0.5 }}>

  Lead ID: {editingRow?.leadID || "N/A"}

</Typography>

</Box>

</Box>
```

```
<Box display="flex" alignItems="center" gap={2}>  
  <Chip  
    label={isEditMode ? "EDIT MODE" : "VIEW MODE"}  
    size="small"  
    sx={{  
      fontWeight: 700,  
      fontSize: "0.75rem",  
      background: isEditMode ? "#fbbf24" : "#60a5fa",  
      color: isEditMode ? "#1f2937" : "#ffffff",  
    }}  
  />  
  <IconButton  
    onClick={handleEditCancel}  
    sx={{ color: "#ffffff", "&:hover": { backgroundColor: "rgba(255,255,255,0.1)" } }}  
  >  
    <CloseRounded />  
  </IconButton>  
</Box>  
</DialogTitle>
```

{/* CONTENT - TABULAR MATRIX FORMAT */}

```
<DialogContent  
  sx={{  
    background: "#f8fafc",  
    p: 0,  
    maxHeight: "calc(90vh - 130px)",  
    overflowY: "auto",  
    overflowX: "hidden",
```

```
"&::webkit-scrollbar": { width: "8px" },
"&::webkit-scrollbar-track": { background: "#e2e8f0" },
"&::webkit-scrollbar-thumb": { background: "#94a3b8", borderRadius: "4px" },
"@media (max-height: 800px)": { maxHeight: "calc(85vh - 130px)" },
}

>

<Box sx={{ p: 2.5 }}>
/* TENDER INFORMATION SECTION */

<Box sx={{ mb: 3 }}>
<Typography
variant="subtitle1"
sx={{
fontWeight: 800,
color: "#1e3a5f",
mb: 2,
display: "flex",
alignItems: "center",
gap: 1,
fontSize: "0.95rem",
textTransform: "uppercase",
letterSpacing: "0.5px",
}}>
</Box>
<Box sx={{ width: 4, height: 20, background: "#1e40af", borderRadius: 1 }} />
Tender Information
</Typography>
<Box
sx={{
```

```
display: "grid",
gridTemplateColumns: { xs: "1fr", sm: "repeat(2, 1fr)", lg: "repeat(4, 1fr)" },
gap: 1.5,
}}
>
{[
{ label: "Lead ID", key: "leadID" },
{ label: "Issue Date", key: "issueDate", isDate: true },
{ label: "Tender Name", key: "tenderName" },
{ label: "Organisation", key: "organisation" },
{ label: "Document Type", key: "documentType" },
{ label: "Tender Type", key: "tenderType" },
].map((field) => (
<Box
key={field.key}
sx={{
background: "#ffffff",
border: "1px solid #e0e7ff",
borderRadius: 2,
p: 2,
"&:hover": { borderColor: "#60a5fa", boxShadow: "0 4px 12px rgba(96,165,250,0.1)" },
transition: "all 0.2s ease",
}}
>
<Typography
variant="caption"
sx={{ color: "#64748b", fontWeight: 600, textTransform: "uppercase",
letterSpacing: "0.5px", fontSize: "0.7rem" }}
```

```
>

{field.label}

</Typography>

{isEditMode ? (
  <TextField
    value={editingRow?.[field.key] || ""}
    onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
    fullWidth
    size="small"
    type={field.isDate ? "date" : "text"}
    InputLabelProps={field.isDate ? { shrink: true } : undefined}
    sx={{
      mt: 1,
      "& .MuiOutlinedInput-root": {
        borderRadius: 1.5,
        background: "#ffffff",
        "& fieldset": { borderColor: "#60a5fa" },
        "&:hover fieldset": { borderColor: "#1e40af" },
        "&.Mui-focused fieldset": { borderColor: "#1e40af", borderWidth: 2 },
      },
      "& .MuiOutlinedInput-input": { color: "#1e293b", fontWeight: 600 },
    }}
  />
): (
  <Typography sx={{ mt: 1, color: "#1e293b", fontWeight: 600, fontSize: "0.95rem" }}>
    {editingRow?.[field.key] || "-"}
  </Typography>
)
```

```
        )}

      </Box>

    ))}

  </Box>

</Box>

/* FINANCIAL DETAILS SECTION */

<Box sx={{ mb: 3 }}>

  <Typography
    variant="subtitle1"
    sx={{
      fontWeight: 800,
      color: "#1e3a5f",
      mb: 2,
      display: "flex",
      alignItems: "center",
      gap: 1,
      fontSize: "0.95rem",
      textTransform: "uppercase",
      letterSpacing: "0.5px",
    }}
  >

    <Box sx={{ width: 4, height: 20, background: "#1e40af", borderRadius: 1 }} />

    Financial Details

  </Typography>

<Box
  sx={{
    display: "grid",

```

```
gridTemplateColumns: { xs: "1fr", sm: "repeat(2, 1fr)", lg: "repeat(2, 1fr)" },
gap: 1.5,
}}
>
{[
{ label: "EMD (Cr)", key: "emdInCrore" },
{ label: "Approx Tender Value (Cr)", key: "approxTenderValueCrore" },
].map((field) => (
<Box
key={field.key}
sx={{{
background: "#ffffff",
border: "1px solid #e0e7ff",
borderRadius: 2,
p: 2,
"&:hover": { borderColor: "#60a5fa", boxShadow: "0 4px 12px rgba(96,165,250,0.1)" },
transition: "all 0.2s ease",
}}}
>
<Typography variant="caption" sx={{ color: "#64748b", fontWeight: 600, textTransform: "uppercase", letterSpacing: "0.5px", fontSize: "0.7rem" }}>
{field.label}
</Typography>
{isEditMode ? (
<TextField
value={editingRow?.[field.key] || ""}
onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
fullWidth
```

```
size="small"

sx={{

  mt: 1,

  "& .MuiOutlinedInput-root": { borderRadius: 1.5, background: "#ffffff", "& fieldset": { borderColor: "#60a5fa" }, "&:hover fieldset": { borderColor: "#1e40af" }, "&.Mui-focused fieldset": { borderColor: "#1e40af", borderWidth: 2 } },

  "& .MuiOutlinedInput-input": { color: "#1e293b", fontWeight: 600 },
}

}>

/>


): (


<Typography sx={{ mt: 1, color: "#1e293b", fontWeight: 600, fontSize: "0.95rem" }}>

{editingRow?.[field.key] || "-"}</Typography>

)}


</Box>

))}</Box></Box>

/* TIMELINE SECTION */

<Box sx={{ mb: 3 }}>

<Typography

variant="subtitle1"

sx={{

  fontWeight: 800,
  color: "#1e3a5f",
  mb: 2,
  display: "flex",
}}>
```

```
    alignItems: "center",
    gap: 1,
    fontSize: "0.95rem",
    textTransform: "uppercase",
    letterSpacing: "0.5px",
  }}
>
<Box sx={{ width: 4, height: 20, background: "#1e40af", borderRadius: 1 }} />
Timeline
</Typography>
<Box
  sx={{
    display: "grid",
    gridTemplateColumns: { xs: "1fr", sm: "repeat(2, 1fr)", lg: "repeat(2, 1fr)" },
    gap: 1.5,
  }}
>
{[
  { label: "Last Date of Submission", key: "lastDateSubmission", isDate: true },
  { label: "Pre-Bid Date & Time", key: "preBidDate", isDate: true },
  { label: "Team Assigned", key: "teamAssigned" },
].map((field) => (
  <Box
    key={field.key}
    sx={{
      background: "#ffffff",
      border: "1px solid #e0e7ff",
      borderRadius: 2,
```

```
p: 2,  
  "&:hover": { borderColor: "#60a5fa", boxShadow: "0 4px 12px  
  rgba(96,165,250,0.1)" },  
  transition: "all 0.2s ease",  
}  
>  
<Typography variant="caption" sx={{ color: "#64748b", fontWeight: 600,  
textTransform: "uppercase", letterSpacing: "0.5px", fontSize: "0.7rem" }}>  
{field.label}  
</Typography>  
{isEditMode ? (  
  <TextField  
    value={editingRow?.[field.key] || ""}  
    onChange={(e) => handleEditFieldChange(field.key, e.target.value)}  
    fullWidth  
    size="small"  
    type={field.isDate ? "datetime-local" : "text"}  
    InputLabelProps={field.isDate ? { shrink: true } : undefined}  
    sx={{  
      mt: 1,  
      "& .MuiOutlinedInput-root": { borderRadius: 1.5, background: "#ffffff", "&  
      fieldset": { borderColor: "#60a5fa" }, "&:hover fieldset": { borderColor: "#1e40af" },  
      "&.Mui-focused fieldset": { borderColor: "#1e40af", borderWidth: 2 } },  
      "& .MuiOutlinedInput-input": { color: "#1e293b", fontWeight: 600 },  
    }}  
  />  
) : (  
  <Typography sx={{ mt: 1, color: "#1e293b", fontWeight: 600, fontSize: "0.95rem"  
}}>
```

```
{editingRow?.[field.key] || "-"}  
</Typography>  
)}  
</Box>  
))}  
</Box>  
</Box>  
  
/* ADDITIONAL INFORMATION SECTION */  
<Box sx={{ mb: 2 }}>  
  <Typography  
    variant="subtitle1"  
    sx={{  
      fontWeight: 800,  
      color: "#1e3a5f",  
      mb: 2,  
      display: "flex",  
      alignItems: "center",  
      gap: 1,  
      fontSize: "0.95rem",  
      textTransform: "uppercase",  
      letterSpacing: "0.5px",  
    }}  
  >  
  <Box sx={{ width: 4, height: 20, background: "#1e40af", borderRadius: 1 }} />  
  Additional Information  
</Typography>  
<Box
```

```
sx={{

  display: "grid",

  gridTemplateColumns: { xs: "1fr", sm: "repeat(2, 1fr)", lg: "repeat(2, 1fr)" },

  gap: 1.5,

}}

>

{[

  { label: "Remarks", key: "remarks", multiline: true },

  { label: "Corrigendum Info", key: "corrigendumInfo", multiline: true },

].map((field) => (

<Box

  key={field.key}

  sx={{

    background: "#ffffff",

    border: "1px solid #e0e7ff",

    borderRadius: 2,

    p: 2,

    "&:hover": { borderColor: "#60a5fa", boxShadow: "0 4px 12px rgba(96,165,250,0.1)" },

    transition: "all 0.2s ease",

  }}

>

<Typography variant="caption" sx={{ color: "#64748b", fontWeight: 600, textTransform: "uppercase", letterSpacing: "0.5px", fontSize: "0.7rem" }}>

  {field.label}

</Typography>

{isEditMode ? (

<TextField

  value={editingRow?.[field.key] || ""}


```

```
        onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
```

```
        fullWidth
```

```
        size="small"
```

```
        multiline={field.multiline}
```

```
        minRows={field.multiline ? 2 : 1}
```

```
        sx={{
```

```
          mt: 1,
```

```
          "& .MuiOutlinedInput-root": { borderRadius: 1.5, background: "#ffffff", "& fieldset": { borderColor: "#60a5fa" }, "&:hover fieldset": { borderColor: "#1e40af" }, "&.Mui-focused fieldset": { borderColor: "#1e40af", borderWidth: 2 } },
```

```
          "& .MuiOutlinedInput-input": { color: "#1e293b", fontWeight: 600 },
```

```
        }}
```

```
      />
```

```
    ) : (
```

```
      <Typography sx={{ mt: 1, color: "#1e293b", fontWeight: 600, fontSize: "0.95rem", whiteSpace: "pre-wrap" }}>
```

```
        {editingRow?.[field.key] || "-"}  
      </Typography>
```

```
    )}
```

```
  </Box>
```

```
)}}
```

```
</Box>
```

```
</Box>
```

```
</Box>
```

```
</DialogContent>
```



```
/* DIALOG ACTIONS */
```

```
<DialogActions sx={{ background: "#f8fafc", borderTop: "1px solid #e0e7ff", p: 2.5, gap: 1.5 }}>
```

```
{!isEditMode ? (
  <>
    <Button onClick={handleEditCancel} sx={{ color: "#64748b", fontWeight: 700,
textTransform: "uppercase", fontSize: "0.85rem", letterSpacing: "0.5px", "&:hover": {
backgroundColor: "#e2e8f0" } }}>
      Close
    </Button>

    <Button onClick={handleEnterEditMode} variant="contained" sx={{ background:
"linear-gradient(135deg, #1e40af 0%, #1e3a5f 100%)", color: "#ffffff", fontWeight: 700,
textTransform: "uppercase", fontSize: "0.85rem", letterSpacing: "0.5px", px: 3, "&:hover": {
background: "linear-gradient(135deg, #1e3a5f 0%, #162e4a 100%)", boxShadow: "0
8px 24px rgba(30,64,95,0.3)" }, "&:active": { transform: "scale(0.98)" } }}>
       Edit Details
    </Button>
  </>
): (
  <>
    <Button onClick={handleCancelEdit} sx={{ color: "#64748b", fontWeight: 700,
textTransform: "uppercase", fontSize: "0.85rem", letterSpacing: "0.5px", "&:hover": {
backgroundColor: "#e2e8f0" } }}>
      Cancel
    </Button>

    <Button onClick={handleEditSave} variant="contained" sx={{ background: "linear-
gradient(135deg, #10b981 0%, #059669 100%)", color: "#ffffff", fontWeight: 700,
textTransform: "uppercase", fontSize: "0.85rem", letterSpacing: "0.5px", px: 3, "&:hover": {
background: "linear-gradient(135deg, #059669 0%, #047857 100%)", boxShadow: "0 8px
24px rgba(16,185,129,0.3)" }, "&:active": { transform: "scale(0.98)" } }}>
       Save Changes
    </>
)
}

✓ What changed:
```

- Removed old fields and replaced with your requested 13 fields.
 - Placed fields logically into Tender Information, Financial Details, Timeline, and Additional Info sections.
 - Dates use proper type="date" or type="datetime-local".
 - Multi-line fields (remarks, corrigendumInfo) are now multiline TextFields.
 - View/Edit toggle remains intact with consistent blue theme styling.
-

If you want, I can also **update the Confirmation Dialog** to show these **specific new fields with a summary before saving** so it fully reflects your updated data structure.

Do you want me to do that next?

Order Received

Got it! You want me to update the Dialog to include this **new set of fields** instead of the previous ones. I'll map them into logical sections (Contract Info, Customer Info, Financial, Additional Info) while keeping the **professional blue theme**, edit/view modes, and multi-line support where appropriate.

Here's the fully updated code snippet with your new fields:

```
{/* EDIT DIALOG - VIEW MODE & EDIT MODE - PROFESSIONAL BLUE THEME */}

<Dialog

  open={editDialogOpen}

  onClose={handleEditCancel}

  maxWidth="xl"

  fullWidth

  PaperProps={{

    sx: {

      borderRadius: 3,

      overflow: "hidden",

      background: "#ffffff",

      boxShadow: "0 25px 50px rgba(0,0,0,0.15), 0 10px 30px rgba(30,64,95,0.2)",

      maxHeight: "90vh",

    },

  }}

  BackdropProps={{

    sx: {

      backdropFilter: "blur(5px)",

      backgroundColor: "rgba(0, 0, 0, 0.4)",

    },

  }}

>
```

```
/* HEADER */

<DialogTitle
  sx={{
    fontWeight: 800,
    display: "flex",
    alignItems: "center",
    justifyContent: "space-between",
    pr: 2,
    background: "linear-gradient(135deg, #1e3a5f 0%, #2d5a8c 100%)",
    color: "#ffffff",
    borderBottom: "3px solid #60a5fa",
    py: 2.5,
  }}
>

<Box display="flex" alignItems="center" gap={2}>
  <Box sx={{ fontSize: 28, fontWeight: 800 }}> 📋 </Box>
  <Box>
    <Typography variant="h6" sx={{ fontWeight: 800, color: "#ffffff" }}>
      {editingRow?.contractName || "Contract Details"}
    </Typography>
    <Typography variant="caption" sx={{ color: "#bfdbfe", mt: 0.5 }}>
      Customer: {editingRow?.customerName || "N/A"}
    </Typography>
  </Box>
</Box>

<Box display="flex" alignItems="center" gap={2}>
  <Chip>
```

```
label={isEditMode ? "EDIT MODE" : "VIEW MODE"}  
size="small"  
sx={{  
  fontWeight: 700,  
  fontSize: "0.75rem",  
  background: isEditMode ? "#fbbf24" : "#60a5fa",  
  color: isEditMode ? "#1f2937" : "#ffffff",  
}}  
/>  
<IconButton  
  onClick={handleEditCancel}  
  sx={{ color: "#ffffff", "&:hover": { backgroundColor: "rgba(255,255,255,0.1)" } }}  
>  
  <CloseRounded />  
</IconButton>  
</Box>  
</DialogTitle>  
  
/* CONTENT */  
<DialogContent  
  sx={{  
    background: "#f8fafc",  
    p: 0,  
    maxHeight: "calc(90vh - 130px)",  
    overflowY: "auto",  
    overflowX: "hidden",  
    "&::-webkit-scrollbar": { width: "8px" },  
    "&::-webkit-scrollbar-track": { background: "#e2e8f0" },  
  }}>
```

```
"&::webkit-scrollbar-thumb": { background: "#94a3b8", borderRadius: "4px" },  
"@media (max-height: 800px)": { maxHeight: "calc(85vh - 130px)" },  
}  
>  
<Box sx={{ p: 2.5 }}>  
/* CONTRACT INFORMATION */  
<Box sx={{ mb: 3 }}>  
<Typography  
variant="subtitle1"  
sx={{  
fontWeight: 800,  
color: "#1e3a5f",  
mb: 2,  
display: "flex",  
alignItems: "center",  
gap: 1,  
fontSize: "0.95rem",  
textTransform: "uppercase",  
letterSpacing: "0.5px",  
}}>  
>  
<Box sx={{ width: 4, height: 20, background: "#1e40af", borderRadius: 1 }} />  
Contract Information  
</Typography>  
<Box  
sx={{  
display: "grid",  
gridTemplateColumns: { xs: "1fr", sm: "repeat(2, 1fr)", lg: "repeat(4, 1fr)" },  
}}>
```

```
gap: 1.5,  
}  
>  
{[  
 { label: "Contract Name", key: "contractName" },  
 { label: "Customer Name", key: "customerName" },  
 { label: "Customer Address", key: "customerAddress", multiline: true },  
 { label: "Order Received Date", key: "orderReceivedDate", isDate: true },  
 { label: "Purchase Order", key: "purchaseOrder" },  
 { label: "Type of Tender", key: "typeOfTender" },  
 { label: "DOM / Export", key: "Dom_or_Export" },  
 ].map((field) => (  
 <Box  
 key={field.key}  
 sx={{  
 background: "#ffffff",  
 border: "1px solid #e0e7ff",  
 borderRadius: 2,  
 p: 2,  
 "&:hover": { borderColor: "#60a5fa", boxShadow: "0 4px 12px  
 rgba(96,165,250,0.1)" },  
 transition: "all 0.2s ease",  
 }  
>  
 <Typography variant="caption" sx={{ color: "#64748b", fontWeight: 600,  
 textTransform: "uppercase", letterSpacing: "0.5px", fontSize: "0.7rem" }}>  
 {field.label}  
 </Typography>  
 {isEditMode ? (
```

```

<TextField
  value={editingRow?.[field.key] || ""}
  onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
  fullWidth
  size="small"
  type={field.isDate ? "date" : "text"}
  multiline={field.multiline || false}
  minRows={field.multiline ? 2 : 1}
  InputLabelProps={field.isDate ? { shrink: true } : undefined}
  sx={{
    mt: 1,
    "& .MuiOutlinedInput-root": { borderRadius: 1.5, background: "#ffffff", "& fieldset": { borderColor: "#60a5fa" }, "&:hover fieldset": { borderColor: "#1e40af" }, "&.Mui-focused fieldset": { borderColor: "#1e40af", borderWidth: 2 } },
    "& .MuiOutlinedInput-input": { color: "#1e293b", fontWeight: 600 },
  }}
/>
): (
  <Typography sx={{ mt: 1, color: "#1e293b", fontWeight: 600, fontSize: "0.95rem", whiteSpace: field.multiline ? "pre-wrap" : "normal" }}>
    {editingRow?.[field.key] || "-"}
  </Typography>
)
}

</Box>
))}

</Box>

</Box>

/* FINANCIAL DETAILS */

```

```
<Box sx={{ mb: 3 }}>  
  <Typography  
    variant="subtitle1"  
    sx={{  
      fontWeight: 800,  
      color: "#1e3a5f",  
      mb: 2,  
      display: "flex",  
      alignItems: "center",  
      gap: 1,  
      fontSize: "0.95rem",  
      textTransform: "uppercase",  
      letterSpacing: "0.5px",  
    }}  
  >  
  <Box sx={{ width: 4, height: 20, background: "#1e40af", borderRadius: 1 }} />  
  Financial Details  
  </Typography>  
  <Box  
    sx={{  
      display: "grid",  
      gridTemplateColumns: { xs: "1fr", sm: "repeat(2, 1fr)", lg: "repeat(2, 1fr)" },  
      gap: 1.5,  
    }}  
  >  
  {[  
    { label: "Value Without GST", key: "valueWithoutGST" },  
    { label: "Value With GST", key: "valueWithGST" },  
  ]}
```

```
].map((field) => (
  <Box
    key={field.key}
    sx={{
      background: "#ffffff",
      border: "1px solid #e0e7ff",
      borderRadius: 2,
      p: 2,
      "&:hover": { borderColor: "#60a5fa", boxShadow: "0 4px 12px rgba(96,165,250,0.1)" },
      transition: "all 0.2s ease",
    }}
  >
  <Typography variant="caption" sx={{ color: "#64748b", fontWeight: 600, textTransform: "uppercase", letterSpacing: "0.5px", fontSize: "0.7rem" }}>
    {field.label}
  </Typography>
  {isEditMode ? (
    <TextField
      value={editingRow?.[field.key] || ""}
      onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
      fullWidth
      size="small"
      sx={{
        mt: 1,
        "& .MuiOutlinedInput-root": { borderRadius: 1.5, background: "#ffffff", "& fieldset": { borderColor: "#60a5fa" }, "&:hover fieldset": { borderColor: "#1e40af" }, "&.Mui-focused fieldset": { borderColor: "#1e40af", borderWidth: 2 } },
        "& .MuiOutlinedInput-input": { color: "#1e293b", fontWeight: 600 },
      }}
    >
  
```

```
        }  
      />  
    ) : (  
      <Typography sx={{ mt: 1, color: "#1e293b", fontWeight: 600, fontSize: "0.95rem" }}>  
    )}>  
      {editingRow?.[field.key] || "-"}  
    </Typography>  
  )}  
  </Box>  
)})  
</Box>  
</Box>  
  
/* ADDITIONAL INFORMATION */  
<Box sx={{ mb: 2 }}>  
  <Typography  
    variant="subtitle1"  
    sx={{  
      fontWeight: 800,  
      color: "#1e3a5f",  
      mb: 2,  
      display: "flex",  
      alignItems: "center",  
      gap: 1,  
      fontSize: "0.95rem",  
      textTransform: "uppercase",  
      letterSpacing: "0.5px",  
    }}  
  >
```

```
>

<Box sx={{ width: 4, height: 20, background: "#1e40af", borderRadius: 1 }} />

Additional Information

</Typography>

<Box

sx={{

display: "grid",

gridTemplateColumns: { xs: "1fr", sm: "repeat(2, 1fr)", lg: "repeat(3, 1fr)" },

gap: 1.5,

}

}>

{[

{ label: "Competitors Info (JSON)", key: "JSON_competitors", multiline: true },

{ label: "Remarks", key: "remarks", multiline: true },

{ label: "Contract Copy", key: "contractCopy" },

{ label: "File Name", key: "fileName" },

{ label: "File Path", key: "filePath" },

{ label: "Hard Disk File Name", key: "hardDiskFileName" },

].map((field) => (

<Box

key={field.key}

sx={{

background: "#ffffff",

border: "1px solid #e0e7ff",

borderRadius: 2,

p: 2,

"&:hover": { borderColor: "#60a5fa", boxShadow: "0 4px 12px

rgba(96,165,250,0.1)" },




```

```
        transition: "all 0.2s ease",
    }
>

<Typography variant="caption" sx={{ color: "#64748b", fontWeight: 600,
textTransform: "uppercase", letterSpacing: "0.5px", fontSize: "0.7rem" }}>
    {field.label}
</Typography>
{isEditMode ? (
    <TextField
        value={editingRow?.[field.key] || ""}
        onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
        fullWidth
        size="small"
        multiline={field.multiline || false}
        minRows={field.multiline ? 2 : 1}
        sx={{
            mt: 1,
            "& .MuiOutlinedInput-root": { borderRadius: 1.5, background: "#ffffff", "& fieldset": { borderColor: "#60a5fa" }, "&:hover fieldset": { borderColor: "#1e40af" }, "&.Mui-focused fieldset": { borderColor: "#1e40af", borderWidth: 2 } },
            "& .MuiOutlinedInput-input": { color: "#1e293b", fontWeight: 600 },
        }}
    />
): (
    <Typography sx={{ mt: 1, color: "#1e293b", fontWeight: 600, fontSize: "0.95rem",
whiteSpace: field.multiline ? "pre-wrap" : "normal" }}>
        {editingRow?.[field.key] || "-"}
    </Typography>
)
}
```

```
</Box>
    )})
</Box>
</Box>
</Box>
</DialogContent>

{/* ACTIONS */}

<DialogActions sx={{ background: "#f8fafc", borderTop: "1px solid #e0e7ff", p: 2.5, gap: 1.5 }}>
  {!isEditMode ? (
    <>
      <Button onClick={handleEditCancel} sx={{ color: "#64748b", fontWeight: 700, textTransform: "uppercase", fontSize: "0.85rem", letterSpacing: "0.5px", "&:hover": { backgroundColor: "#e2e8f0" } }}>
        Close
      </Button>
      <Button onClick={handleEnterEditMode} variant="contained" sx={{ background: "linear-gradient(135deg, #1e40af 0%, #1e3a5f 100%)", color: "#ffffff", fontWeight: 700, textTransform: "uppercase", fontSize: "0.85rem", letterSpacing: "0.5rem", px: 3, "&:hover": { background: "linear-gradient(135deg, #1e3a5f 0%, #162e4a 100%)", boxShadow: "0 8px 24px rgba(30,64,95,0.3)" }, "&:active": { transform: "scale(0.98)" } }}>
         Edit Details
      </Button>
    </>
  ):(
    <>
      <Button onClick={handleCancelEdit} sx={{ color: "#64748b", fontWeight: 700, textTransform: "uppercase", fontSize: "0.85rem", "&:hover": { backgroundColor: "#e2e8f0" } }}>
        Cancel
      </Button>
    </>
  )
</>

```

```

        </Button>

        <Button onClick={handleEditSave} variant="contained" sx={{ background: "linear-
gradient(135deg, #10b981 0%, #059669 100%)", color: "#ffffff", fontWeight: 700,
textTransform: "uppercase", fontSize: "0.85rem", px: 3, "&:hover": { background: "linear-
gradient(135deg, #059669 0%, #047857 100%)", boxShadow: "0 8px 24px
rgba(16,185,129,0.3)" }, "&:active": { transform: "scale(0.98)" } }}>
 Save Changes
</Button>
</>
)}
</DialogActions>
</Dialog>

```

 **Notes on changes:**

- Sections: **Contract Information, Financial Details, Additional Information.**
 - Added **DOM/Export**, JSON competitors, multi-line support for remarks and address.
 - File-related fields included: contractCopy, fileName, filePath, hardDiskFileName.
 - Dates (orderReceivedDate) and numeric values (valueWithoutGST, valueWithGST) formatted as input fields accordingly.
 - Maintained **edit/view mode toggle** and professional blue gradient theme.
-

If you want, I can **also regenerate the Confirmation Dialog** for this new data structure so users can preview these exact fields before saving.

Do you want me to do that?

Perfect! You want the dialog updated for this **new tender-related set of fields**. I'll organize them into **logical sections** like Tender Information, Customer Info, Financial Details, Additional Info, etc., while keeping the **professional blue theme** and the **view/edit toggle**.

Here's the fully updated code:

```
/* EDIT DIALOG - VIEW MODE & EDIT MODE - PROFESSIONAL BLUE THEME */  
  
<Dialog  
  open={editDialogOpen}  
  onClose={handleEditCancel}  
  maxWidth="xl"  
  fullWidth  
  PaperProps={{  
    sx: {  
      borderRadius: 3,  
      overflow: "hidden",  
      background: "#ffffff",  
      boxShadow: "0 25px 50px rgba(0,0,0,0.15), 0 10px 30px rgba(30,64,95,0.2)",  
      maxHeight: "90vh",  
    },  
  }}  
  BackdropProps={{  
    sx: {  
      backdropFilter: "blur(5px)",  
      backgroundColor: "rgba(0, 0, 0, 0.4)",  
    },  
  }}  
>  
/* HEADER */  
<DialogTitle
```

```
sx={{  
    fontWeight: 800,  
    display: "flex",  
    alignItems: "center",  
    justifyContent: "space-between",  
    pr: 2,  
    background: "linear-gradient(135deg, #1e3a5f 0%, #2d5a8c 100%)",  
    color: "#ffffff",  
    borderBottom: "3px solid #60a5fa",  
    py: 2.5,  
}}  
>  
<Box display="flex" alignItems="center" gap={2}>  
  <Box sx={{ fontSize: 28, fontWeight: 800 }}>  </Box>  
  <Box>  
    <Typography variant="h6" sx={{ fontWeight: 800, color: "#ffffff" }}>  
      {editingRow?.tenderName || "Tender Details"}  
    </Typography>  
    <Typography variant="caption" sx={{ color: "#bfdbfe", mt: 0.5 }}>  
      Reference: {editingRow?.tenderReferenceNo || "N/A"}  
    </Typography>  
  </Box>  
</Box>  
  
<Box display="flex" alignItems="center" gap={2}>  
  <Chip  
    label={isEditMode ? "EDIT MODE" : "VIEW MODE"}  
    size="small"
```

```
sx={{  
    fontWeight: 700,  
    fontSize: "0.75rem",  
    background: isEditMode ? "#fbff24" : "#60a5fa",  
    color: isEditMode ? "#1f2937" : "#ffffff",  
}}  
>  
<IconButton  
    onClick={handleEditCancel}  
    sx={{ color: "ffffff", "&:hover": { backgroundColor: "rgba(255,255,255,0.1)" } }}  
>  
    <CloseRounded />  
</IconButton>  
</Box>  
</DialogTitle>  
  
 {/* CONTENT */}  
<DialogContent  
    sx={{  
        background: "#f8fafc",  
        p: 0,  
        maxHeight: "calc(90vh - 130px)",  
        overflowY: "auto",  
        overflowX: "hidden",  
        "&::-webkit-scrollbar": { width: "8px" },  
        "&::-webkit-scrollbar-track": { background: "#e2e8f0" },  
        "&::-webkit-scrollbar-thumb": { background: "#94a3b8", borderRadius: "4px" },  
        "@media (max-height: 800px)": { maxHeight: "calc(85vh - 130px)" },  
    }}>
```

```
    }}

>

<Box sx={{ p: 2.5 }}>
  {/* TENDER INFORMATION */}

<Box sx={{ mb: 3 }}>
  <Typography
    variant="subtitle1"
    sx={{
      fontWeight: 800,
      color: "#1e3a5f",
      mb: 2,
      display: "flex",
      alignItems: "center",
      gap: 1,
      fontSize: "0.95rem",
      textTransform: "uppercase",
      letterSpacing: "0.5px",
    }}
  >

  <Box sx={{ width: 4, height: 20, background: "#1e40af", borderRadius: 1 }} />
  Tender Information
  </Typography>

<Box
  sx={{
    display: "grid",
    gridTemplateColumns: { xs: "1fr", sm: "repeat(2, 1fr)", lg: "repeat(3, 1fr)" },
    gap: 1.5,
  }}>
```

```

>
[{"label": "Tender Name", "key": "tenderName"}, {"label": "Tender Reference No", "key": "tenderReferenceNo"}, {"label": "Tender Type", "key": "tenderType"}, {"label": "Document Type", "key": "documentType"}].map((field) => (
<Box
  key={field.key}
  sx={{background: "#ffffff", border: "1px solid #e0e7ff", borderRadius: 2, p: 2, "&:hover": {borderColor: "#60a5fa", boxShadow: "0 4px 12px rgba(96,165,250,0.1)"}, transition: "all 0.2s ease",}}
>
<Typography variant="caption" sx={{color: "#64748b", fontWeight: 600, textTransform: "uppercase", letterSpacing: "0.5px", fontSize: "0.7rem"}}>
  {field.label}
</Typography>
{isEditMode ? (
<TextField
  value={editingRow?.[field.key] || ""}
  onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
  fullWidth
  size="small"
)
: (
<Text
  value={field.value}
  onClick={() => handleEditFieldChange(field.key, "")}
  style={{border: "1px solid #e0e7ff", padding: 5px, cursor: "pointer", width: "100%"}}>
  {field.value}
</Text>
)
}

```

```
sx={{  
    mt: 1,  
  
    "& .MuiOutlinedInput-root": { borderRadius: 1.5, background: "#ffffff", "&  
fieldset": { borderColor: "#60a5fa" }, "&:hover fieldset": { borderColor: "#1e40af" },  
"&.Mui-focused fieldset": { borderColor: "#1e40af", borderWidth: 2 } },  
  
    "& .MuiOutlinedInput-input": { color: "#1e293b", fontWeight: 600 },  
}  
}}  
/>  
) : (  
  <Typography sx={{ mt: 1, color: "#1e293b", fontWeight: 600, fontSize: "0.95rem"  
}}>  
  {editingRow?.[field.key] || "-"}  
  </Typography>  
)  
</Box>  
))}  
</Box>  
</Box>  
  
/* CUSTOMER INFORMATION */  
<Box sx={{ mb: 3 }}>  
  <Typography  
    variant="subtitle1"  
    sx={{  
      fontWeight: 800,  
      color: "#1e3a5f",  
      mb: 2,  
      display: "flex",  
      alignItems: "center",  
    }}>
```

```
gap: 1,  
fontSize: "0.95rem",  
textTransform: "uppercase",  
letterSpacing: "0.5px",  
}  
>  
<Box sx={{ width: 4, height: 20, background: "#1e40af", borderRadius: 1 }} />  
Customer Information  
</Typography>  
<Box  
sx={{  
display: "grid",  
gridTemplateColumns: { xs: "1fr", sm: "repeat(2, 1fr)", lg: "repeat(2, 1fr)" },  
gap: 1.5,  
}  
>  
{[  
{ label: "Customer Name", key: "customerName" },  
{ label: "Customer Address", key: "customerAddress", multiline: true },  
].map((field) => (  
<Box  
key={field.key}  
sx={{  
background: "#ffffff",  
border: "1px solid #e0e7ff",  
borderRadius: 2,  
p: 2,
```

```

    "&:hover": { borderColor: "#60a5fa", boxShadow: "0 4px 12px
      rgba(96,165,250,0.1)" },
      transition: "all 0.2s ease",
    }
  >

  <Typography variant="caption" sx={{ color: "#64748b", fontWeight: 600,
  textTransform: "uppercase", letterSpacing: "0.5px", fontSize: "0.7rem" }}>
    {field.label}
  </Typography>

  {isEditMode ? (
    <TextField
      value={editingRow?.[field.key] || ""}
      onChange={(e) => handleEditFieldChange(field.key, e.target.value)}
      fullWidth
      size="small"
      multiline={field.multiline || false}
      minRows={field.multiline ? 2 : 1}
      sx={{
        mt: 1,
        "& .MuiOutlinedInput-root": { borderRadius: 1.5, background: "#ffffff", "&
        fieldset": { borderColor: "#60a5fa" }, "&:hover fieldset": { borderColor: "#1e40af" },
        "&.Mui-focused fieldset": { borderColor: "#1e40af", borderWidth: 2 } },
        "& .MuiOutlinedInput-input": { color: "#1e293b", fontWeight: 600 },
      }}
    />
  ) : (
    <Typography sx={{ mt: 1, color: "#1e293b", fontWeight: 600, fontSize: "0.95rem",
    whiteSpace: field.multiline ? "pre-wrap" : "normal" }}>
      {editingRow?.[field.key] || "-"}
    </Typography>
  )
}

```

```
</Typography>
)}
</Box>
))}

</Box>
</Box>

{/* FINANCIAL DETAILS */}

<Box sx={{ mb: 3 }}>
<Typography
variant="subtitle1"
sx={{
fontWeight: 800,
color: "#1e3a5f",
mb: 2,
display: "flex",
alignItems: "center",
gap: 1,
fontSize: "0.95rem",
textTransform: "uppercase",
letterSpacing: "0.5px",
}}>
>
<Box sx={{ width: 4, height: 20, background: "#1e40af", borderRadius: 1 }} />
Financial Details
</Typography>
<Box
sx={{
```

```
display: "grid",
gridTemplateColumns: { xs: "1fr", sm: "repeat(2, 1fr)", lg: "repeat(2, 1fr)" },
gap: 1.5,
}}
>
{[
 { label: "Value in Cr (w/o GST)", key: "valueInCrWithoutGST" },
 { label: "Value in Cr (with GST)", key: "valueInCrWithGST" },
 { label: "Quoted Price", key: "quotedPrice" },
 { label: "Technical Score", key: "technicalScore" },
].map((field) => (
<Box
key={field.key}
sx={{{
background: "#ffffff",
border: "1px solid #e0e7ff",
borderRadius: 2,
p: 2,
"&:hover": { borderColor: "#60a5fa", boxShadow: "0 4px 12px rgba(96,165,250,0.1)" },
transition: "all 0.2s ease",
}}}
>
<Typography variant="caption" sx={{ color: "#64748b", fontWeight: 600, textTransform: "uppercase", letterSpacing: "0.5px", fontSize: "0.7rem" }}>
{field.label}
</Typography>
{isEditMode ? (
<TextField
```

```
        value={editingRow?.[field.key] || ""}

        onChange={(e) => handleEditFieldChange(field.key, e.target.value)}

        fullWidth

        size="small"

        sx={{

            mt: 1,

            "& .MuiOutlinedInput-root": { borderRadius: 1.5, background: "#ffffff", "& fieldset": { borderColor: "#60a5fa" }, "&:hover fieldset": { borderColor: "#1e40af" }, "&.Mui-focused fieldset": { borderColor: "#1e40af", borderWidth: 2 } },

            "& .MuiOutlinedInput-input": { color: "#1e293b", fontWeight: 600 },

        }}

        />

    ) : (

        <Typography sx={{ mt: 1, color: "#1e293b", fontWeight: 600, fontSize: "0.95rem" }}>

            {editingRow?.[field.key] || "-"}

        </Typography>

    )}

    </Box>

))}

</Box>

</Box>

/* ADDITIONAL INFORMATION */

<Box sx={{ mb: 2 }}>

<Typography

variant="subtitle1"

sx={{

fontWeight: 800,
```

```
        color: "#1e3a5f",
        mb: 2,
        display: "flex",
        alignItems: "center",
        gap: 1,
        fontSize: "0.95rem",
        textTransform: "uppercase",
        letterSpacing: "0.5px",
    })
>
<Box sx={{ width: 4, height: 20, background: "#1e40af", borderRadius: 1 }} />
Additional Information
</Typography>
<Box
sx={{
    display: "grid",
    gridTemplateColumns: { xs: "1fr", sm: "repeat(2, 1fr)", lg: "repeat(3, 1fr)" },
    gap: 1.5,
}}
>
{[
    { label: "Reason for Losing", key: "reasonForLossing", multiline: true },
    { label: "Year We Lost", key: "yearWeLost" },
    { label: "Partners", key: "partners" },
    { label: "Competitors", key: "competitors", multiline: true },
].map((field) => (
    <Box
        key={field.key}
```

```
sx={{  
    background: "#ffffff",  
    border: "1px solid #e0e7ff",  
    borderRadius: 2,  
    p: 2,  
    "&:hover": { borderColor: "#60a5fa", boxShadow: "0 4px 12px  
    rgba(96,165,250,0.1)" },  
    transition: "all 0.2s ease",  
}}  
>  
<Typography variant="caption" sx={{ color: "#64748b", fontWeight: 600,  
textTransform: "uppercase", letterSpacing: "0.5px", fontSize: "0.7rem" }}>  
{field.label}  
</Typography>  
{isEditMode ? (  
    <TextField  
        value={editingRow?.[field.key] || ""}  
        onChange={(e) => handleEditFieldChange(field.key, e.target.value)}  
        fullWidth  
        size="small"  
        multiline={field.multiline || false}  
        minRows={field.multiline ? 2 : 1}  
        sx={{  
            mt: 1,  
            "& .MuiOutlinedInput-root": { borderRadius: 1.5, background: "#ffffff", "&  
            fieldset": { borderColor: "#60a5fa" }, "&:hover fieldset": { borderColor: "#1e40af" },  
            "&.Mui-focused fieldset": { borderColor: "#1e40af", borderWidth: 2 },,  
            "& .MuiOutlinedInput-input": { color: "#1e293b", fontWeight: 600 },  
        }}  
    >
```

```

        />

      ) : (
        <Typography sx={{ mt: 1, color: "#1e293b", fontWeight: 600, fontSize: "0.95rem", whiteSpace: field.multiline ? "pre-wrap" : "normal" }}>
          {editingRow?.[field.key] || "-"}
        </Typography>
      )}

    </Box>
  ))}

</Box>

</Box>

</Box>

</DialogContent>

/* ACTIONS */

<DialogActions sx={{ background: "#f8fafc", borderTop: "1px solid #e0e7ff", p: 2.5, gap: 1.5 }}>
  {!isEditMode ? (
    <>
      <Button onClick={handleEditCancel} sx={{ color: "#64748b", fontWeight: 700, textTransform: "uppercase", fontSize: "0.85rem", letterSpacing: "0.5px", "&:hover": { backgroundColor: "#e2e8f0" } }}>
        Close
      </Button>
      <Button onClick={handleEnterEditMode} variant="contained" sx={{ background: "linear-gradient(135deg, #1e40af 0%, #1e3a5f 100%)", color: "#ffffff", fontWeight: 700, textTransform: "uppercase", fontSize: "0.85rem", letterSpacing: "0.5rem", px: 3, "&:hover": { background: "linear-gradient(135deg, #1e3a5f 0%, #162e4a 100%)", boxShadow: "0 8px 24px rgba(30,64,95,0.3)" }, "&:active": { transform: "scale(0.98)" } }}>
        Edit Details
      </Button>
    </>
  ) : (
    <Button onClick={handleEditCancel} sx={{ color: "#64748b", fontWeight: 700, textTransform: "uppercase", fontSize: "0.85rem", letterSpacing: "0.5px", "&:hover": { backgroundColor: "#e2e8f0" } }}>
      Close
    </Button>
  )}
</DialogActions>

```

```

        </Button>
      </>
    ) : (
      <>
        <Button onClick={handleCancelEdit} sx={{ color: "#64748b", fontWeight: 700,
textTransform: "uppercase", fontSize: "0.85rem", "&:hover": { backgroundColor:
"#e2e8f0" } }}>
          Cancel
        </Button>
        <Button onClick={handleEditSave} variant="contained" sx={{ background: "linear-
gradient(135deg, #10b981 0%, #059669 100%)", color: "#ffffff", fontWeight: 700,
textTransform: "uppercase", fontSize: "0.85rem", px: 3, "&:hover": { background: "linear-
gradient(135deg, #059669 0%, #047857 100%)", boxShadow: "0 8px 24px
rgba(16,185,129,0.3)" }, "&:active": { transform: "scale(0.98)" } }}>
           Save Changes
        </Button>
      </>
    )
  </DialogActions>
</Dialog>

```

Highlights for this version:

- **

Sections**: Tender Info | Customer Info | Financial Details | Additional Info

- **Editable & view-only mode** with professional blue theme.
 - **Multiline support** for address, reason for losing, and competitors.
 - All **numeric / financial fields** separated (Value in Cr, Quoted Price, Technical Score).
-

I can also create a **summary/confirmation dialog** for this tender that shows all these fields neatly before saving—very useful for audits or reviews.

Do you want me to generate that too?