**Detailed Step-by-Step Guide to Building the Room Booking Canvas App in Power Apps**

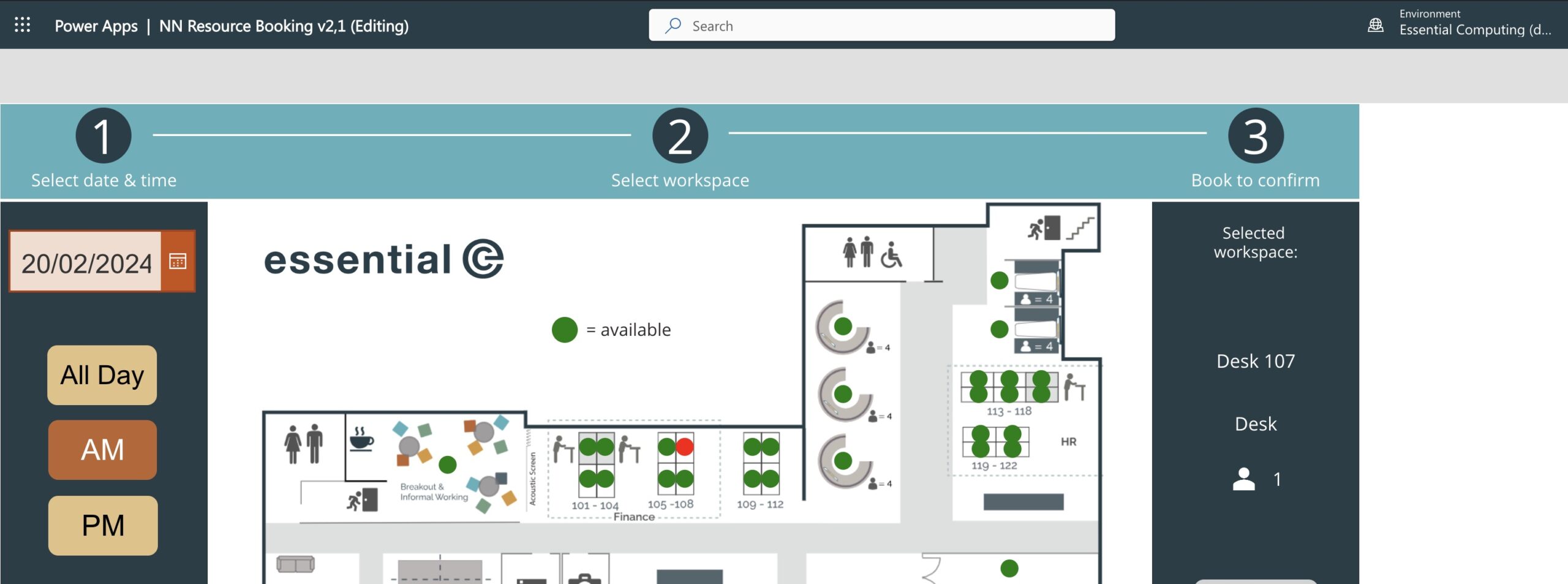
Since the YAML snippets aren't working due to schema issues, I'll provide a complete, visual step-by-step guide to build the app manually in Power Apps Studio. This covers **all controls**, **Power Fx formulas** (code), and **exact steps** to meet your requirements: SharePoint Online data storage, active/inactive rooms, 30-minute minimum bookings from 6 AM to 8 PM, time zone agnostic handling (using UTC), availability checks with errors, a dialog for viewing bookings, displaying today's bookings on the home screen, and cancellation functionality.

This guide assumes you have access to Power Apps (make.powerapps.com) and a SharePoint Online site. The app is built for a phone layout but can be adjusted for tablet. I'll include screenshots of similar screens for reference (from Power Apps examples; your app will look comparable after building).

**Step 1: Set Up SharePoint Lists**

Before building the app, create the data sources in SharePoint.

1. Go to your SharePoint site (e.g., <https://yourtenant.sharepoint.com/sites/YourSite>).
2. Click **New > List** to create two lists:
   * **Rooms List**:
     + Columns:
       - Title (Single line of text, required) – Room name, e.g., "Conference Room A".
       - Active (Yes/No, default Yes) – To toggle room visibility.
     + Add sample data: Create 3-5 rooms, set some to Active=No.
   * **Bookings List**:
     + Columns:
       - Room (Lookup to Rooms list, allow multiple=No, required) – Links to room.
       - StartDateTime (Date and time, include time=Yes, required) – Booking start (stored in UTC).
       - EndDateTime (Date and time, include time=Yes, required) – Booking end (UTC).
       - BookedBy (Person or Group, required) – Who booked it.
       - Status (Choice, options: "Active", "Cancelled"; default="Active").
     + Add sample bookings for testing.
3. Note: SharePoint stores dates in UTC automatically, making it time zone agnostic when queried with Power Apps' UTC functions.



[essential.co.uk](https://www.essential.co.uk/blog/articles/using-power-apps-to-create-desk-booking-system/)

Example of setting up SharePoint lists in Power Apps for a booking system.

**Step 2: Create the Blank Canvas App**

1. Go to make.powerapps.com > **Create** > **Blank app** > **Blank canvas app** (choose Phone layout for simplicity).
2. Name the app "RoomBookingApp" and save it.
3. Add data sources:
   * In the left pane, click **Data** > **Add data** > Search for "SharePoint" > Connect to your site > Select "Rooms" and "Bookings" lists.
   * Add **Office 365 Users** connector: **Add data** > Search "Office 365 Users".
4. Rename the default screen (Screen1) to "scrHome" (right-click > Rename).

**Step 3: Set App-Level Variables and OnStart**

1. In the left pane, select **App** (top of Tree view).
2. In the formula bar, set **OnStart** to this Power Fx code (initializes variables and time slots):

text

Set(varCurrentUser, Office365Users.MyProfileV2());

Set(varShowDialog, false);

Set(varSelectedRoom, Blank());

Set(varSelectedDate, Blank());

Set(varNewStart, Blank());

Set(varNewEnd, Blank());

Set(varConflicts, Blank());

Set(varAvailable, false);

Set(varTimeSlots, ForAll(Sequence(29), {Time: DateAdd(Time(6,0,0), (Value-1)\*30, Minutes), Display: Text(DateAdd(Time(6,0,0), (Value-1)\*30, Minutes), "hh:mm AM/PM")}));

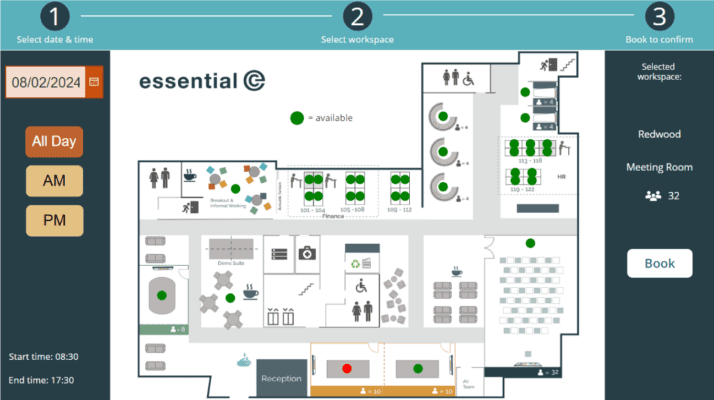
* + This creates 29 time slots from 6:00 AM to 8:00 PM (e.g., "6:00 AM", "6:30 AM", ..., "8:00 PM").

1. Run OnStart (click the ... menu on App > Run OnStart) to test.

**Step 4: Build the Home Screen (scrHome)**

This screen shows the user's bookings for today.

1. Add a **Label** control (Insert > Label):
   * Name: lblWelcome
   * Text: "Welcome, " & varCurrentUser.DisplayName & ". Your bookings today:"
   * Position: Top of screen (X=20, Y=20), Font size=18.
2. Add a **Vertical gallery** (Insert > Gallery > Vertical):
   * Name: galMyBookings
   * Items: Sort(Filter(Bookings, BookedBy.Email = varCurrentUser.Mail, DateValue(StartDateTime) = DateValue(Now(), "UTC"), Status.Value = "Active"), StartDateTime, Ascending)
     + Filters to active bookings today (UTC for time zone agnostic).
   * Position: Below lblWelcome (Y=lblWelcome.Y + lblWelcome.Height + 10), Width=Parent.Width - 40, Height=Parent.Height - 200.
   * In the gallery template (edit mode):
     + Add Label: Text = ThisItem.Room.Title (room name).
     + Add Label: Text = Text(ThisItem.StartDateTime, "hh:mm AM/PM") (start time).
     + Add Label: Text = Text(ThisItem.EndDateTime, "hh:mm AM/PM") (end time).
     + Add Icon (Insert > Icons > Trash): Name=icnCancel, OnSelect = Patch(Bookings, ThisItem, {Status: {Value: "Cancelled"}}); Notify("Booking cancelled.", NotificationType.Success); Refresh(Bookings)
3. Add a **Button** (Insert > Button):
   * Name: btnBookRoom
   * Text: "Book a Room"
   * OnSelect: Navigate(scrRoomSelection)
   * Position: Bottom center (Y=Parent.Height - Height - 20).



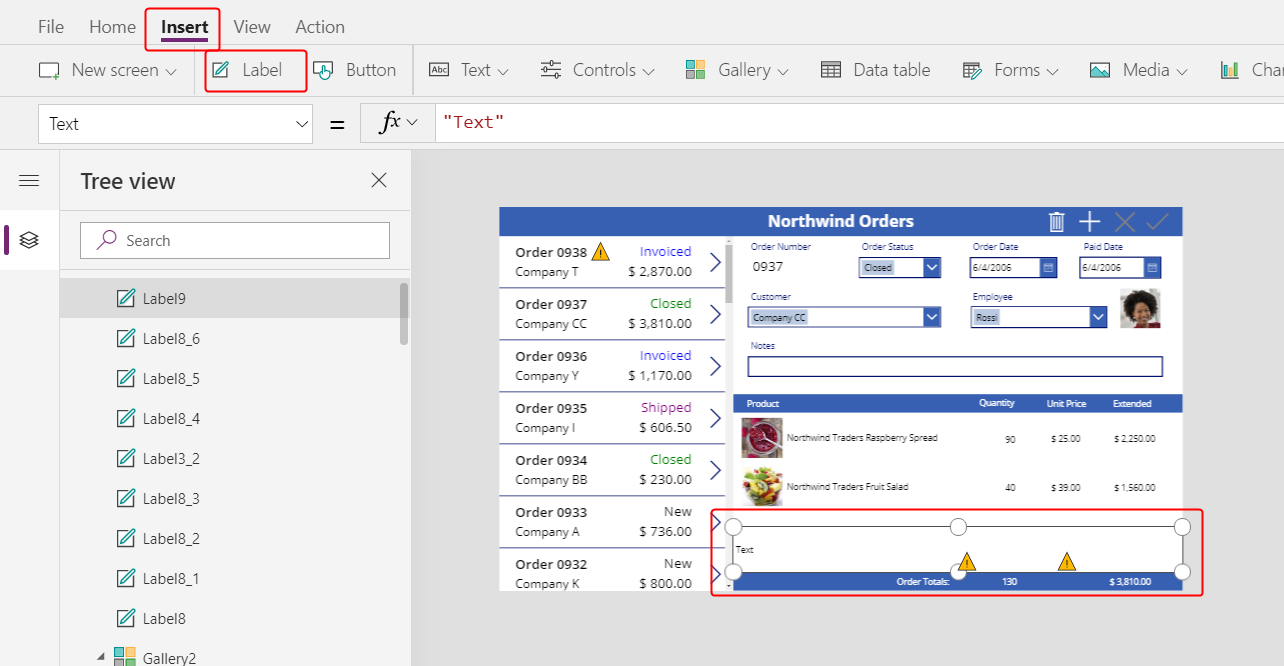
[essential.co.uk](https://www.essential.co.uk/blog/articles/using-power-apps-to-create-desk-booking-system/)

Example home screen with a gallery of bookings.

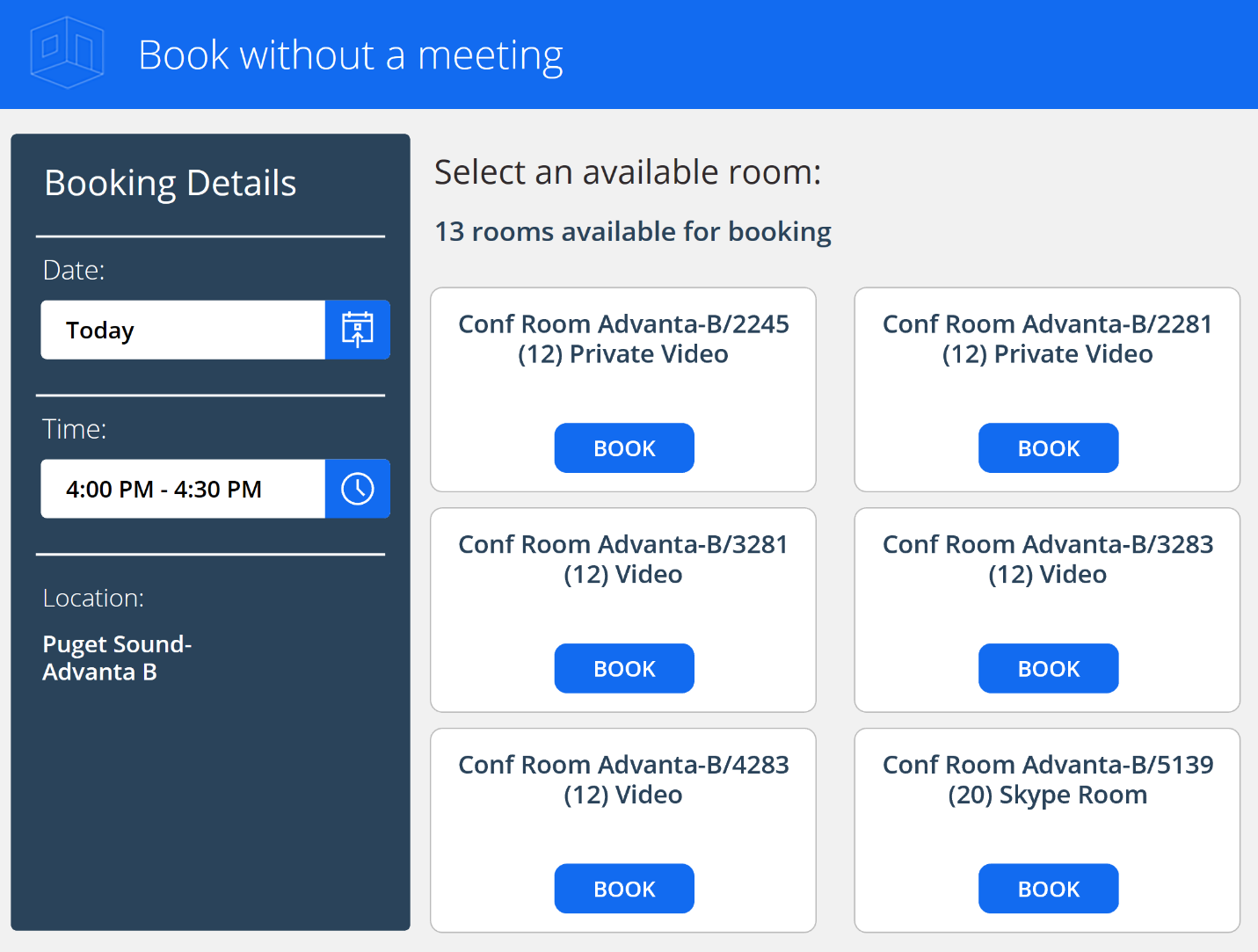
**Step 5: Add and Build the Room Selection Screen (scrRoomSelection)**

Users select active rooms and view bookings via a dialog.

1. Add a new screen: Insert > New screen > Blank > Rename to "scrRoomSelection".
2. Add a **Vertical gallery**:
   * Name: galRooms
   * Items: Filter(Rooms, Active)
     + Only shows active rooms.
   * Position: Top (Y=20), Width=Parent.Width - 40, Height=Parent.Height - 100.
   * In template:
     + Add Label: Text = ThisItem.Title (room name).
     + Add Icon (Calendar): Name=icnViewBookings, OnSelect = Set(varSelectedRoom, ThisItem); Set(varSelectedDate, Today()); Set(varShowDialog, true)
     + Make gallery selectable: OnSelect = Set(varSelectedRoom, ThisItem); Navigate(scrBooking)
3. Simulate a dialog (no built-in modals; use visibility):
   * Add a **Container** (Insert > Layout > Container):
     + Name: conDialog
     + Visible: varShowDialog
     + Fill: RGBA(0,0,0,0.5) (semi-transparent overlay)
     + Width=Parent.Width, Height=Parent.Height.
   * Inside conDialog, add a **Rectangle**:
     + Name: recDialogBg
     + Fill: White
     + Width=Parent.Width \* 0.8, Height=Parent.Height \* 0.6, X=(Parent.Width - Width)/2, Y=(Parent.Height - Height)/2 (centered).
   * Inside conDialog (above recDialogBg in layers), add a **Vertical gallery**:
     + Name: galDialogBookings
     + Items: Sort(Filter(Bookings, Room.Id = varSelectedRoom.Id, DateValue(StartDateTime) = DateValue(varSelectedDate, "UTC"), Status.Value = "Active"), StartDateTime)
     + Position: Inside recDialogBg (X=10, Y=10), Width=recDialogBg.Width - 20, Height=recDialogBg.Height - 60.
     + Template: Labels for ThisItem.BookedBy.DisplayName, Text(ThisItem.StartDateTime, "hh:mm AM/PM"), Text(ThisItem.EndDateTime, "hh:mm AM/PM").
   * Inside conDialog, add a **Button**:
     + Name: btnCloseDialog
     + Text: "Close"
     + OnSelect: Set(varShowDialog, false)
     + Position: Bottom of recDialogBg.
4. Add a back icon/button: OnSelect = Back().



[learn.microsoft.com](https://learn.microsoft.com/en-us/power-apps/maker/canvas-apps/northwind-orders-canvas-part3)



[microsoft.com](https://www.microsoft.com/en-us/power-platform/blog/power-apps/office-template-book-a-room-now-available/)

**Step 6: Add and Build the Booking Screen (scrBooking)**

Users select date/time, check availability, and book.

1. Add a new screen: Rename to "scrBooking".
2. Add a **Label**:
   * Name: lblBookingFor
   * Text: "Booking for: " & varSelectedRoom.Title
   * Position: Top.
3. Add a **Date picker** (Insert > Input > Date picker):
   * Name: dpDate
   * DefaultDate: Today()
   * Position: Below lblBookingFor.
4. Add two **Dropdown** controls (Insert > Input > Drop down):
   * Name: ddStartTime
     + Items: varTimeSlots
     + Default: First(varTimeSlots).Display
     + Position: Below dpDate.
   * Name: ddEndTime
     + Items: varTimeSlots
     + Default: Last(FirstN(varTimeSlots, 2)).Display (30 min after start).
     + Position: Below ddStartTime.
5. Add a **Button**:
   * Name: btnCheckAvailability
   * Text: "Check Availability"
   * OnSelect:

text

Set(varNewStart, DateAdd(dpDate.SelectedDate, TimeValue(ddStartTime.Selected.Display), "UTC"));

Set(varNewEnd, DateAdd(dpDate.SelectedDate, TimeValue(ddEndTime.Selected.Display), "UTC"));

If(varNewEnd <= varNewStart || DateDiff(varNewStart, varNewEnd, Minutes) < 30,

Notify("Invalid: End after start, min 30 min.", NotificationType.Error),

Set(varConflicts, Filter(Bookings, Room.Id = varSelectedRoom.Id, Status.Value = "Active", !(EndDateTime <= varNewStart || StartDateTime >= varNewEnd)));

If(CountRows(varConflicts) > 0,

Notify("Time slot unavailable.", NotificationType.Error),

Set(varAvailable, true); Notify("Available!", NotificationType.Success)

)

)

* + - Checks min 30 min, overlaps (using UTC).
  + Position: Below ddEndTime.

1. Add a **Button**:
   * Name: btnBook
   * Text: "Book"
   * Visible: varAvailable
   * OnSelect:

text

Patch(Bookings, Defaults(Bookings), {

Room: {Id: varSelectedRoom.Id, Value: varSelectedRoom.Title},

StartDateTime: varNewStart,

EndDateTime: varNewEnd,

BookedBy: varCurrentUser,

Status: {Value: "Active"}

});

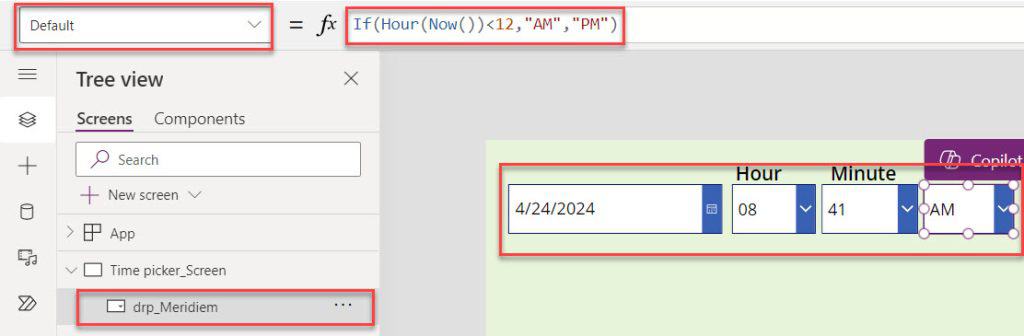
Set(varAvailable, false);

Notify("Booked!", NotificationType.Success);

Navigate(scrHome)

* + Position: Next to btnCheckAvailability.

1. Add an **Icon** (Calendar):
   * Name: icnViewBookings
   * OnSelect: Set(varSelectedDate, dpDate.SelectedDate); Set(varShowDialog, true)
   * Position: Top right (reuses the dialog from scrRoomSelection; ensure conDialog is on this screen or make it a component).
2. Duplicate the conDialog container from scrRoomSelection (copy-paste) for the dialog here.
3. Add a back icon/button: OnSelect = Back().



[spguides.com](https://www.spguides.com/date-time-picker-in-powerapps/)

Example booking screen with date picker and time dropdowns.

**Step 7: Testing, Polish, and Publishing**

1. **Test the App**:
   * Click **Preview** (F5) or run from browser.
   * Create bookings, check overlaps (should error), view dialogs, cancel, switch time zones (dates remain consistent via UTC).
   * Debug formulas in the formula bar or Monitor tool (View > Monitor).
2. **Polish**:
   * Add themes: Select controls > Properties > Color/Fill.
   * Add error labels if needed (e.g., Visible on conflicts).
   * For large data: Ensure delegable functions (Filter/Sort are delegable for SharePoint up to 2000 items).
3. **Publish**: File > Save > Publish > Publish this version.
4. Share the app via Power Apps portal.

This builds the full app. If you encounter formula errors, copy-paste exactly (Power Fx is case-sensitive). For more advanced features (e.g., components for reusable dialog), let me know!