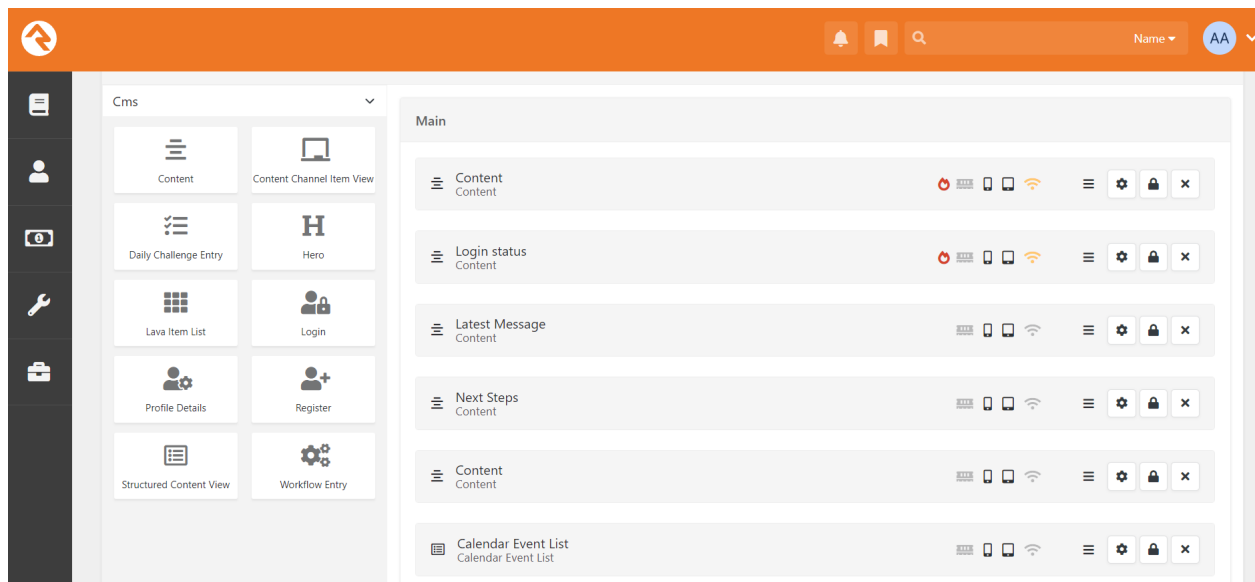
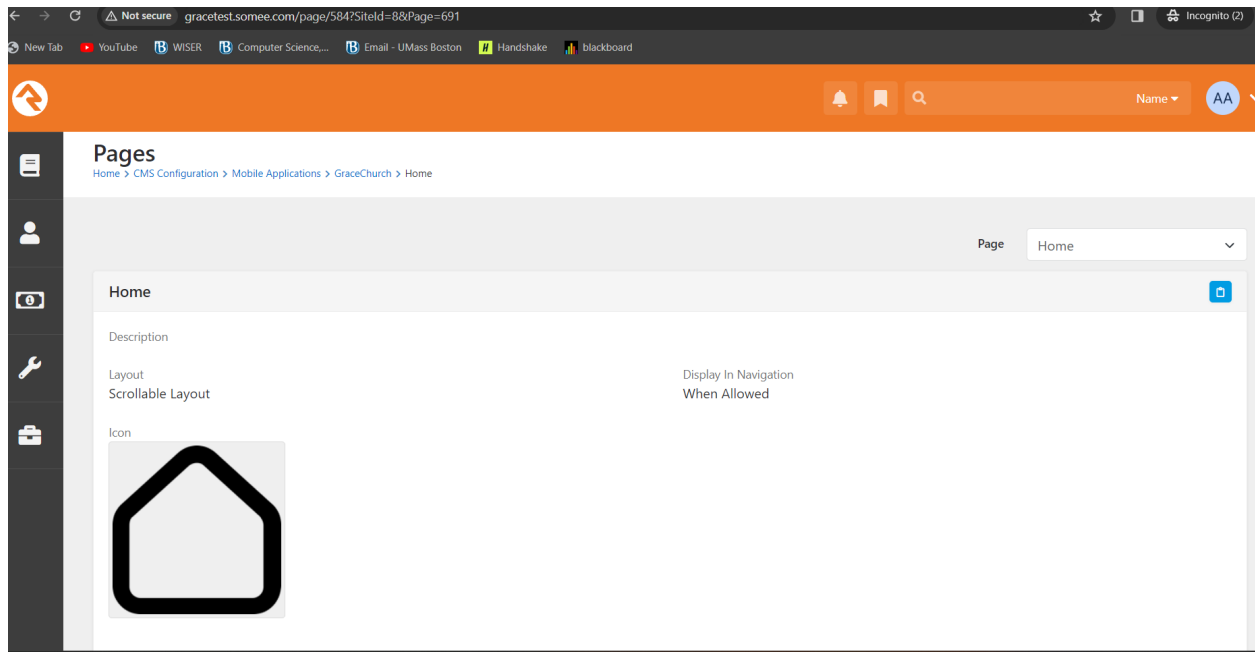


# CODE DOCUMENTATION

## Home Page-



## 1. Content block-

This block displays custom XAML & Lava content on the page. Arguably the most fundamental block in any Rock Mobile application, this block could be described as a necessity, used to render Lava and XAML within your shell, as provided in the "Content" block setting.

### Code-

```
<StackLayout>
  <Grid>

                                                                    <Image
Source="https://assets-global.website-files.com/5f6b9a421d5a61e1d0cd9e3d/62f28
5c4f9aa3441840257d6_nathan-mullet-pmiW630yDPE-unsplash.jpg"
          Aspect="AspectFill"  />

    <BoxView BackgroundColor="#80000000" />

    <Label Text="Prayer"
          TextColor="White"
          FontSize="36"
          FontAttributes="Bold"
          HorizontalOptions="Center"
          VerticalOptions="Center"  />
  </Grid>
</StackLayout>
```

### Explanation-

- <StackLayout>: This is the outermost container. StackLayout arranges its child elements in a single line, which can be oriented vertically (by default) or horizontally. In this case, since there are no specific attributes set, it's using the default vertical orientation.
- <Grid>: Inside the StackLayout, there is a Grid element. A Grid is used for laying out elements in rows and columns, but in this snippet, no rows or columns are explicitly defined, so it behaves as a single container.
- <Image>: This is an image element. It has several attributes:
  - Source: The URL of the image to display, pointing to an external image hosted online.
  - Aspect: Set to AspectFill. This means the image will fill the available space without preserving its aspect ratio. If the aspect ratios of the image and the container differ, the image will be clipped to fit.

- <BoxView>: This creates a simple rectangle of a single color. Here, it's being used as an overlay:
  - BackgroundColor: The color is set to a semi-transparent black (#80000000, where 80 is the alpha channel for 50% transparency, and 000000 is the hex code for black). This overlay can give the text better contrast and readability against the background image.
- <Label>: This is a text element with several styling attributes:
  - Text: The content of the label, set to "Prayer".
  - TextColor: The color of the text, set to white.
  - FontSize: The size of the font, set to 36.
  - FontAttributes: Additional font styling, set to bold.
  - HorizontalOptions and VerticalOptions: These properties are set to Center, meaning the label will be centered both horizontally and vertically within its parent container.

## 2. Profile Bar-

### Code-

```
<StackLayout Margin="0,0,0,0" Orientation="Vertical" VerticalOptions="Center"
BackgroundColor="#7bb446">
    <StackLayout Margin="0,10,0,10" Orientation="Horizontal"
HorizontalOptions="Start">
        <Rock:Icon IconClass="user" StyleClass="text-primary"
Margin="10,0,0,0" TextColor="White" VerticalOptions="CenterAndExpand"
FontSize="24" />
        <Label Grid.Column="1" StyleClass="pl-3" Text="Hi {{
CurrentPerson.FirstName }}" TextColor="White" Margin="0,10,0,10"
VerticalOptions="CenterAndExpand">
            <Label.GestureRecognizers>
                <TapGestureRecognizer Command="{Binding PushPage}"
CommandParameter="09ee7ee8-0b44-4b88-b5ca-4dc26dec006f" />
            </Label.GestureRecognizers>
        </Label>
    </StackLayout>

    <Rock:Divider />
</StackLayout>
```

## Explanation-

- Outer StackLayout:
  - Margin: Set to "0,0,0,0", meaning there is no margin around this layout.
  - Orientation: Set to "Vertical", meaning its child elements will be stacked vertically.
  - VerticalOptions: Set to "Center", which will center the layout vertically within its parent container.
  - BackgroundColor: Set to "#7bb446", a specific shade of green.
- Inner StackLayout (for the user greeting):
  - Margin: Set to "0,10,0,10", adding a 10-unit space on the top and bottom.
  - Orientation: Set to "Horizontal", meaning its children (the icon and label) will be arranged side by side.
  - HorizontalOptions: Set to "Start", aligning its children to the start (left side in a left-to-right layout) of the container.
- Rock:Icon:
  - A custom component (presumably from the RockRMS framework), representing an icon.
  - IconClass: Set to "user", which likely determines the icon's appearance.
  - StyleClass: Set to "text-primary", which would apply predefined styling, likely related to color.
  - Margin: Set to "10,0,0,0", adding a space to the left of the icon.
  - TextColor: Set to "White".
  - VerticalOptions: Set to "CenterAndExpand", centering the icon vertically and allowing it to fill the available space.
  - FontSize: Set to "24".
- Label (for the greeting text):
  - Text: Set to "Hi {{ CurrentPerson.FirstName }}!", which seems to be a template that will be replaced with the user's first name.
  - TextColor: Set to "White".
  - Margin: Set to "0,10,0,10", adding space around the label.
  - VerticalOptions: Set to "CenterAndExpand".
  - Label.GestureRecognizers: Contains a TapGestureRecognizer that seems to execute a command (PushPage) when the label is tapped, with a parameter ("09ee7ee8-0b44-4b88-b5ca-4dc26dec006f"), possibly used to navigate to another page.
- Rock:Divider:
  - A custom component, likely used to insert a divider line or a similar visual separation between elements in the layout.

### 3. Latest Message-

#### Code-

```
<StackLayout StyleClass="p-16" Spacing="16">
  <Label Text="Latest Message"
    FontFamily="Poppins"
    FontAttributes="Bold"
    FontSize="18"
    TextColor="#000000"
    Margin="5,0,5,10"
    HorizontalTextAlignment="Start" />

  <Rock:ContainedCard
    Image="https://faithisland.org/wp-content/uploads/2015/06/Humility.jpg"
    ImageRatio=".75:1">
    <StackLayout Orientation="Horizontal"
      VerticalOptions="FillAndExpand">
      <Label Text="Title goes here"
        FontAttributes="Bold"
        FontFamily="Poppins"
        FontSize="24"
        HorizontalOptions="Start"
        Margin="0,0,20,0" />
      <StackLayout Orientation="Horizontal"
        HorizontalOptions="End">
        <Rock:Icon IconClass="fa-volume-up"
          IconFamily="FontAwesomeSolid"
          FontSize="24"
          Margin="40,0,0,0" />
        <Rock:Icon.GestureRecognizers>
          <TapGestureRecognizer Command="{Binding PlayAudio}"
            CommandParameter="http://www.noiseaddicts.com/samples_1w72b820/2541.mp3" />
        </Rock:Icon.GestureRecognizers>
        </Rock:Icon>
        <Rock:Icon IconClass="fa-play"
          IconFamily="FontAwesomeSolid"
          FontSize="24"
          Margin="40,0,0,0">
          <Rock:Icon.GestureRecognizers>
            <TapGestureRecognizer Command="{Binding PlayVideo}"
              CommandParameter="https://youtu.be/BwGzK4YStW4" />
          </Rock:Icon.GestureRecognizers>
```

```

        </Rock:Icon>
    </StackLayout>
</StackLayout>
    <Rock:ContainedCard.GestureRecognizers>
        <TapGestureRecognizer Command="{Binding OpenBrowser}"

CommandParameter="https://thatsgrace.org/messages/how-to-express-gratitude/"
/>
    </Rock:ContainedCard.GestureRecognizers>
</Rock:ContainedCard>
</StackLayout>

```

## Explanation-

- Outer StackLayout:
  - StyleClass: "p-16", likely a predefined class that adds padding.
  - Spacing: "16", meaning there's a 16-unit space between its child elements.
- Label for "Latest Message":
  - Displays the text "Latest Message".
  - FontFamily: "Poppins", which is a clean and modern font.
  - FontAttributes: "Bold", making the text bold.
  - FontSize: "18".
  - TextColor: "#000000" (black).
  - Margin: "5,0,5,10", adding margin around the label.
  - HorizontalTextAlignment: "Start", aligning the text to the start (left in left-to-right languages).
- Rock:ContainedCard:
  - A custom component, likely used for displaying a card-like layout.
  - Image: A URL pointing to an image, which will be displayed in the card.
  - ImageRatio: ".75:1", defining the aspect ratio of the image.
- Inner StackLayout for Title:
  - Orientation: "Horizontal".
  - VerticalOptions: "FillAndExpand", making it fill the available vertical space.
  - Contains a Label for the title ("Title goes here") with similar styling as the "Latest Message" label, but with a larger font size ("24").
- Nested StackLayout for Icons:
  - Orientation: "Horizontal".
  - HorizontalOptions: "End", aligning the icons to the end (right side).
  - Contains two Rock:Icon elements, each with a TapGestureRecognizer:
    - The first icon (IconClass="fa-volume-up") is for playing audio. Tapping it triggers a PlayAudio command with an audio URL as the parameter.

- The second icon (IconClass="fa-play") is for playing video. Tapping it triggers a PlayVideo command with a video URL as the parameter.
  - Each icon has a margin of "40,0,0,0", spacing them out within the layout.
- Gesture Recognizer on Rock:ContainedCard:
  - A TapGestureRecognizer is attached to the entire ContainedCard.
  - Tapping anywhere on the card triggers the OpenBrowser command, with a URL as the parameter. This suggests that tapping the card will navigate the user to a webpage.

#### 4. Next Steps-

```
<StackLayout StyleClass="p-16" Spacing="16">
  <Label Text="Next Steps"
    FontFamily="Poppins"
    FontAttributes="Bold"
    FontSize="18"
    TextColor="#000000"
    Margin="5,0,5,10"
    HorizontalTextAlignment="Start" />
```

```
<Rock:InlineCard
  Title="Groups"
  HeaderLocation="Top"
  ContentLocation="Bottom"
  TitleJustification="Center"
```

```
Image="https://www.focusonthefamily.com/wp-content/uploads/2021/05/ChurchGathering-AndNetworking-2048x1367.jpg" CardRatio="0.75:1"
```

```
BackgroundMask="Top-Bottom"
BackgroundMaskColor="#000000"
BackgroundMaskOpacity=".6">
```

God has a purpose for your life. The best way to discover it is to begin applying Biblical teachings to your life. That's where Life Groups come in. We offer a variety of in-person, virtual and hybrid life groups!

```
<Rock:InlineCard.GestureRecognizers>
```

```
    <TapGestureRecognizer Command="{Binding PushPage}"
    CommandParameter="7791519d-31bf-4f67-9a50-430ea85c8fba" />
```

```
</Rock:InlineCard.GestureRecognizers>
```

```
</Rock:InlineCard>
```

```
</StackLayout>
```

## Explanation-

- Outer StackLayout:
  - StyleClass: "p-16", which is likely a predefined class that adds padding around the StackLayout.
  - Spacing: "16", meaning there's a 16-unit space between its child elements.
- Label for "Next Steps":
  - Displays the text "Next Steps".
  - FontFamily: "Poppins", a modern and clean font.
  - FontAttributes: "Bold", making the text bold.
  - FontSize: "18".
  - TextColor: "#000000" (black).
  - Margin: "5,0,5,10", adding margin around the label.
  - HorizontalTextAlignment: "Start", aligning the text to the start (left in left-to-right languages).
- Rock:InlineCard Component:
  - A custom component, likely designed to display content in a card-like format.
  - Title: "Groups", indicating the card's title.
  - HeaderLocation: "Top", suggesting the title is located at the top of the card.
  - ContentLocation: "Bottom", indicating that the content is at the bottom part of the card.
  - TitleJustification: "Center", centering the title text.
  - Image: A URL to an image, which will be displayed in the card.
  - CardRatio: "0.75:1", defining the aspect ratio of the card.
  - BackgroundMask: "Top-Bottom", likely applies a mask effect to the background image.
  - BackgroundMaskColor: "#000000" (black), specifying the color of the mask.
  - BackgroundMaskOpacity: ".6", setting the opacity of the mask to 60%.
  - The card contains text describing the purpose of Life Groups and their variety (in-person, virtual, and hybrid).
  - Rock:InlineCard.GestureRecognizers:
    - The gesture recognizer is bound to a PushPage command with a parameter ("7791519d-31bf-4f67-9a50-430ea85c8fba"), likely used to navigate to another page related to the "Groups" when the card is tapped.

## 5. Calendar-Event List-

This block displays a list of events from a calendar.

### Code-

```
<Frame HasShadow="false" StyleClass="calendar-event">
```



```

    <StackLayout Spacing="0">
        <Label StyleClass="calendar-event-title" Text="{Binding Name}"
FontSize="16" />
        <StackLayout Orientation="Horizontal">
            {% if Item.EndDateTime == null %}
                <Label StyleClass="calendar-event-text" Text="{{
Item.StartDateTime | Date:'h:mm tt' }}" LineBreakMode="NoWrap" />
            {% else %}
                <Label StyleClass="calendar-event-text" Text="{{
Item.StartDateTime | Date:'h:mm tt' }}" - "{{ Item.EndDateTime | Date:'h:mm tt'
}}" LineBreakMode="NoWrap" />
            {% endif %}
            <Label HorizontalOptions="FillAndExpand"
StyleClass="calendar-event-audience" Text="{{ Item.Audiences | Select:'Name' |
Join:', ' | Escape }}" />
            <Label StyleClass="calendar-event-campus" Text="{{ Item.Campus |
Escape }}" HorizontalTextAlignment="End" LineBreakMode="NoWrap" />
        </StackLayout>
    </StackLayout>
</Frame>

```

## Explanation-

- <Frame>:
  - HasShadow: Set to "false", indicating that the frame will not display a shadow.
  - StyleClass: "calendar-event", which likely applies predefined styling specific to calendar event displays.
- Outer StackLayout:
  - Spacing: "0", meaning there will be no space between its child elements.
  - This layout serves as the primary container for the event details.
- Event Title Label:
  - StyleClass: "calendar-event-title", applying specific styles to the event title.
  - Text: {Binding Name}, indicating that the text will be bound to a property named Name, likely representing the event's name.
  - FontSize: "16".
- Inner StackLayout (Horizontal Orientation):
  - Contains elements that are displayed horizontally, side by side.
- Conditional Event Time Display:
  - The snippet includes a conditional templating block ({% if ... %} / {% else %}).

- If Item.EndDateTime is null (meaning the event has no end time), it displays only the start time.
- If Item.EndDateTime is not null, it displays both the start and end times.
- The times are formatted using a date filter (Date:'h:mm tt'), likely converting them into a human-readable format like "2:00 PM".
- Event Audience Label:
  - Displays the event's audience.
  - The text is generated by selecting the Name property from Item.Audiences and joining them with a comma.
  - HorizontalOptions: "FillAndExpand", which means the label will expand to fill the available horizontal space.
- Event Campus Label:
  - Displays the campus associated with the event.
  - Text: Binds to Item.Campus, and Escape is used to ensure the text is displayed correctly.
  - HorizontalTextAlignment: "End", aligning the text to the end (right side).
  - LineBreakMode: "NoWrap", preventing the text from wrapping to the next line.

### Day Header Template-

The XAML to use when rendering the day header above a grouping of events.

### Code-

```
<Label      Text="{Binding      .,      StringFormat='{0:dddd      MMMM      d}'}"
StyleClass="calendar-events-day, h2" Margin="5,0,5,10" FontSize="16"/>
```

### Explanation-

- Text Attribute with Data Binding and String Formatting:
  - {Binding ., StringFormat='{0:dddd MMMM d}'}: This is a data-binding expression.
    - Binding . means it binds to the current context or the default property of the data context in scope.
    - StringFormat='{0:dddd MMMM d}' is used to format the bound data.
    - {0:dddd MMMM d} is a custom format string:
      - dddd represents the full name of the day (e.g., Monday, Tuesday).
      - MMMM represents the full name of the month (e.g., January, February).
      - d represents the day of the month.

- The format '{0:dddd MMMM d}' will display dates in a format like "Monday January 1".
- StyleClass Attribute:
  - "calendar-events-day, h2": This applies predefined style classes to the label.
    - calendar-events-day likely adds styles specific to displaying days in a calendar view.
    - h2 suggests a style similar to an HTML h2 heading, which could mean larger font size, bold text, etc.
- Margin Attribute:
  - "5,0,5,10": This sets the margin around the label.
    - 5 units of margin on the left and right sides.
    - 0 units of margin on the top.
    - 10 units of margin on the bottom.
  - This spacing separates the label from other elements or the edges of its container.
- FontSize Attribute:
  - "16": This sets the size of the font used in the label to 16.

## 6. Screenshots-



## Next Steps



### Groups

God has a purpose for your life. The best way to discover it is to begin applying Biblical teachings to your life. That's where Life Groups come in. We offer a variety of in-person, virtual and hybrid life groups!

## Upcoming Events

# December 2023



Monday December 25

**Christmas Prayer**

12:00 AM

Avon



Groups



Prayer



Home



Teams



More

