

# Unit 3 – Low-Code / No-Code Development

Certification: [C\\_LCNC\\_02](#)

Low-Code / No-Code Applications

# First Mini Application with SAP Build Apps

## Demo: Note-Taking application



# Features

- Cross platform – Web, Mobile (iOS, Android)
- Data persistence
- Seamless navigation

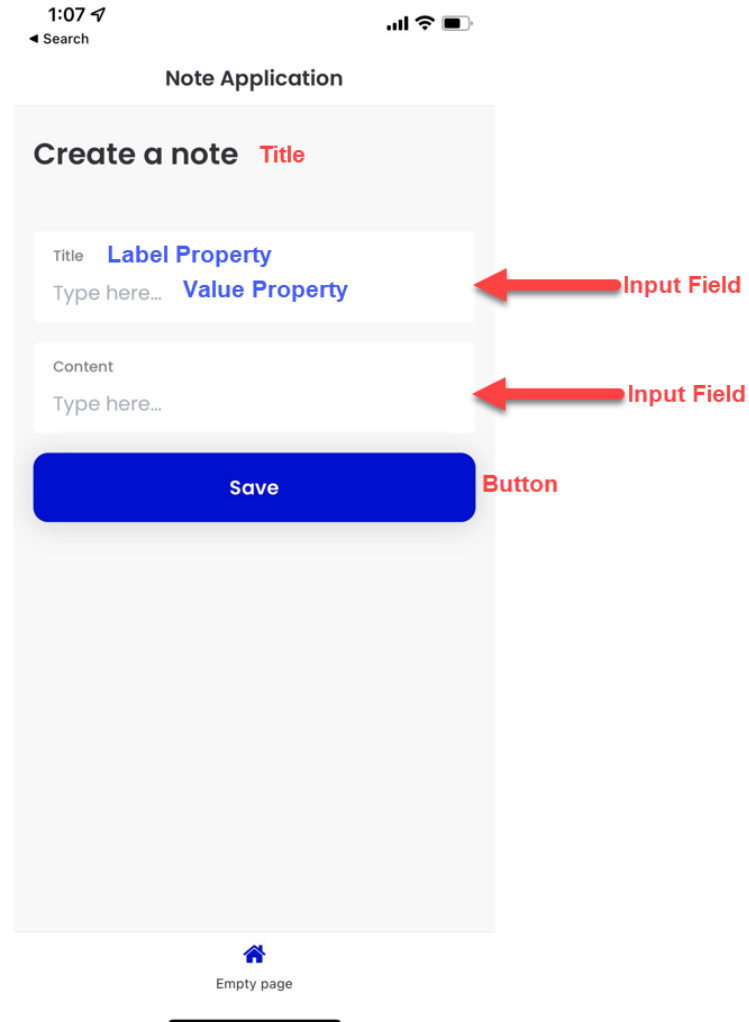
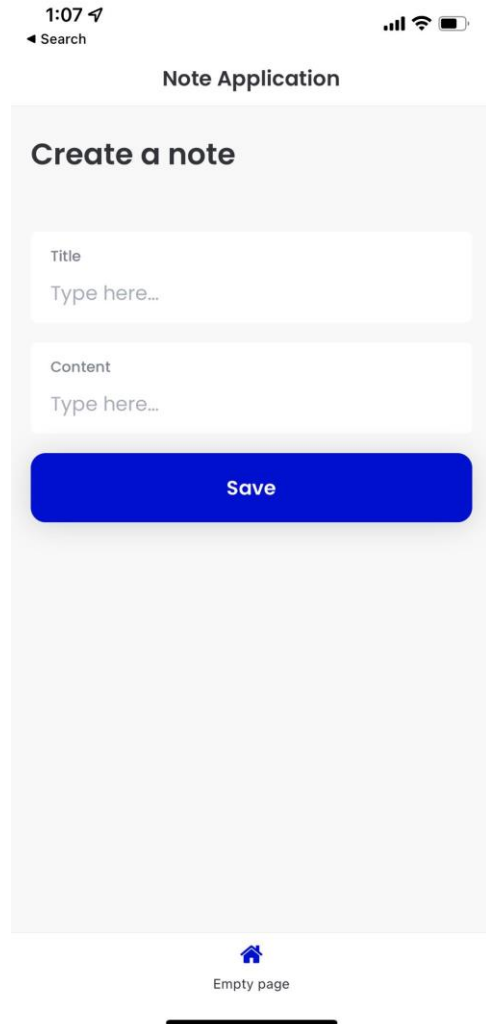
# Steps involved

1. Setting up the pre-requisites
2. Setting up the UI – View (main) and Components
3. Bind variables to UI components
4. Create database for persistence
5. Setting up the UI – View (list) and Components
6. Bind variables to UI components
7. Save notes in database + navigate to list view

# Step 1 – Pre-Requisites

1. Subscribe to [SAP Build Apps](#) via a booster from SAP Business Technology Platform
2. Get the [SAP Build Apps Preview](#) app on your mobile phone

# Step 2 – View (Main) and its components



# Steps involved

1. Setting up the pre-requisites (Completed)
2. Setting up the UI – View (main) and Components (Completed)
3. Bind variables to UI components
4. Create database for persistence
5. Setting up the UI – View (list) and Components
6. Bind variables to UI components
7. Save notes in database + navigate to list view

# Step 3 – Bind variables to UI components

We have 2 Input Fields in the Main View...

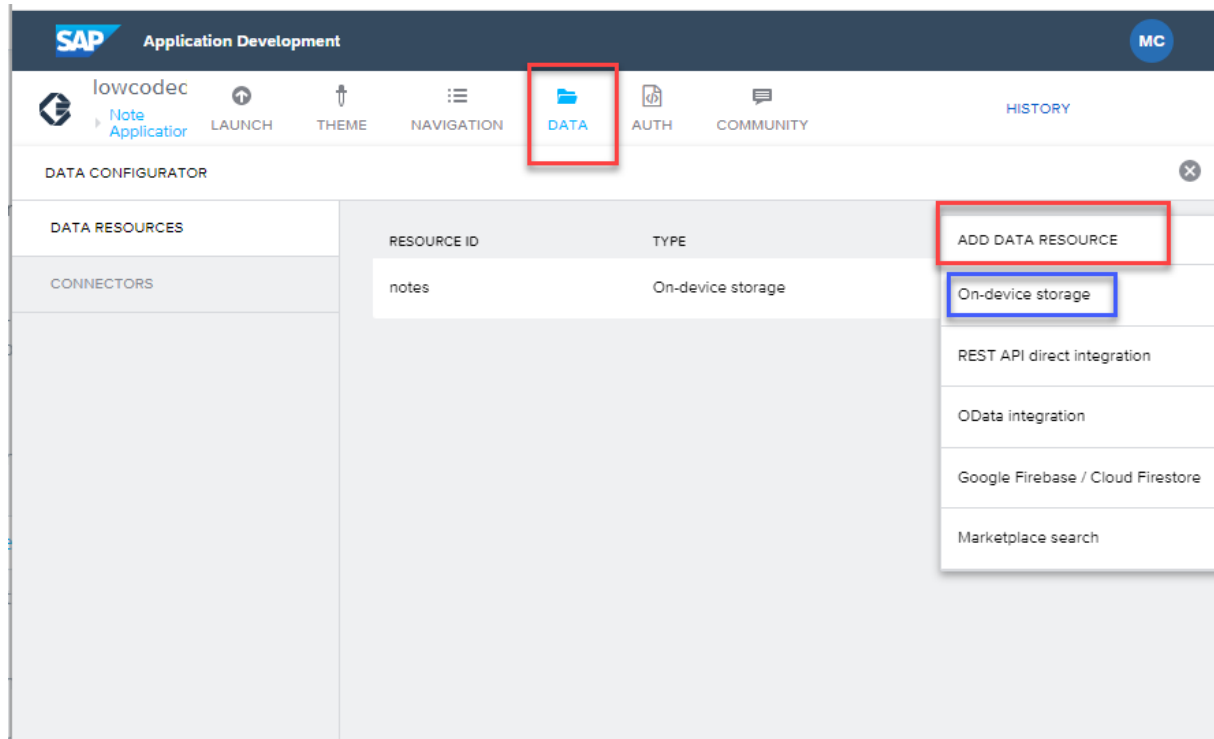
- Title
- Content

Title Input Field (Value property): `title`

Content Input Field (Value property): `content`



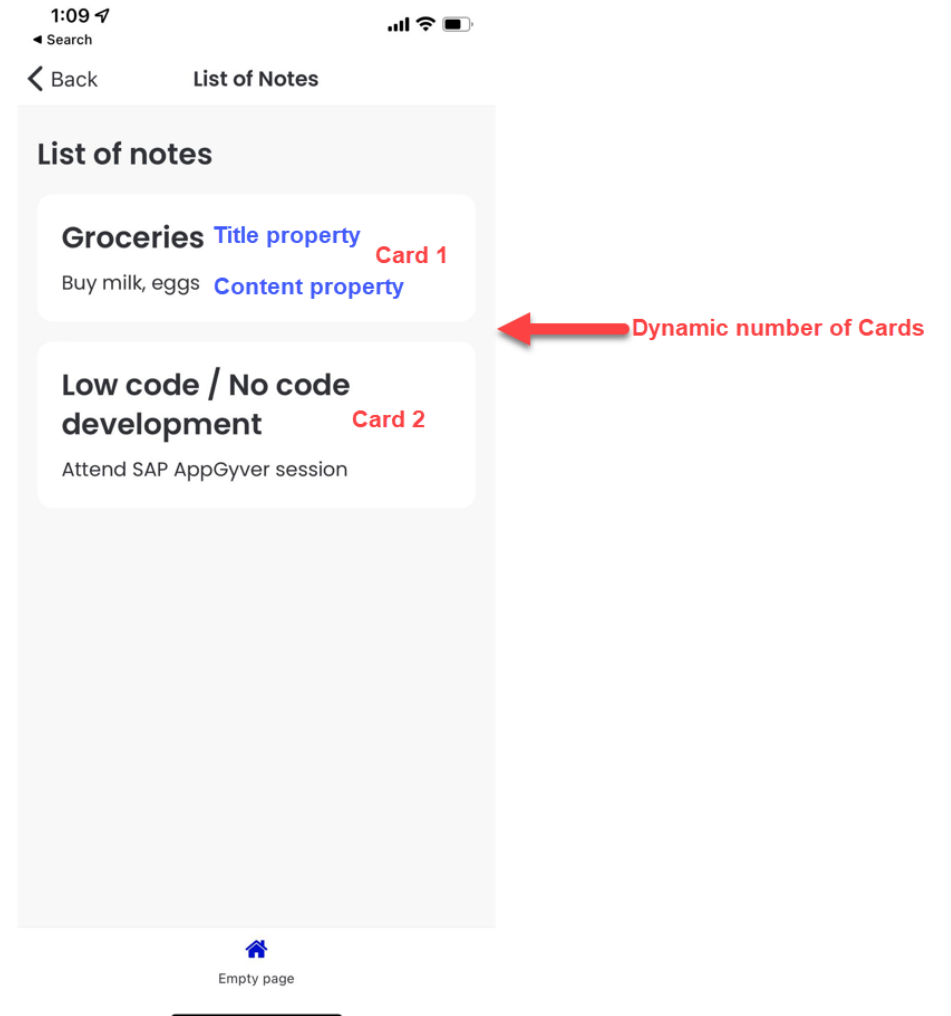
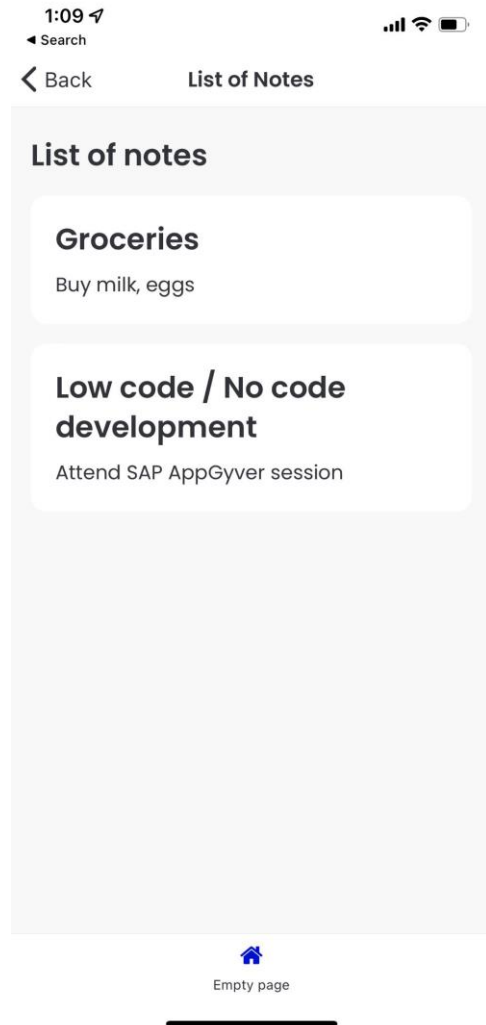
# Step 4 – Create database for persistence



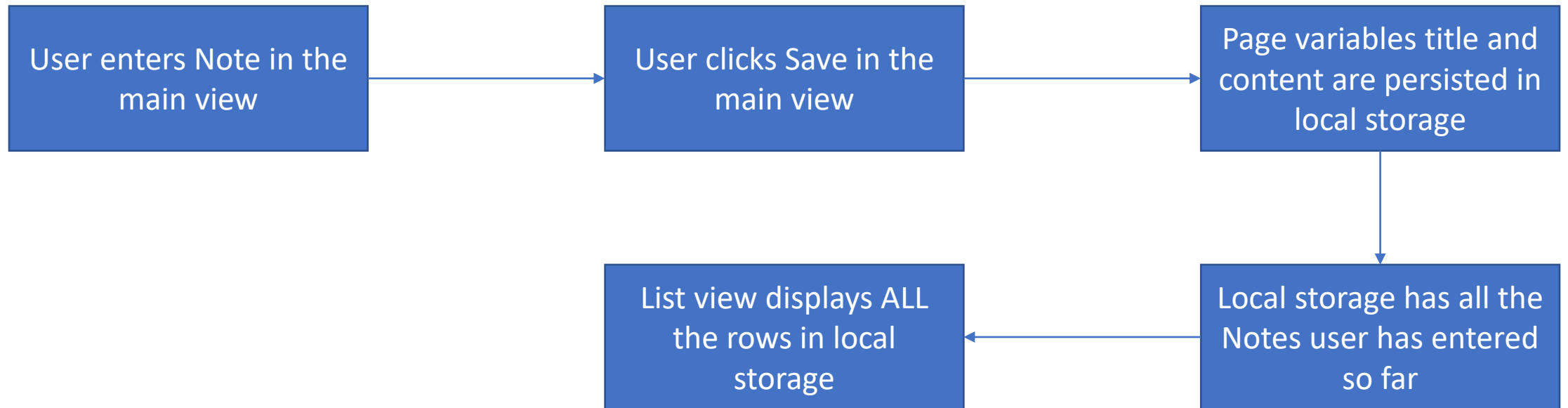
# Steps involved

1. Setting up the pre-requisites (Completed)
2. Setting up the UI – View (main) and Components (Completed)
3. Bind variables to UI components (Completed)
4. Create database for persistence (Completed)
5. Setting up the UI – View (list) and Components
6. Bind variables to UI components
7. Save notes in database + navigate to list view

# Step 5 – View (List) and its components

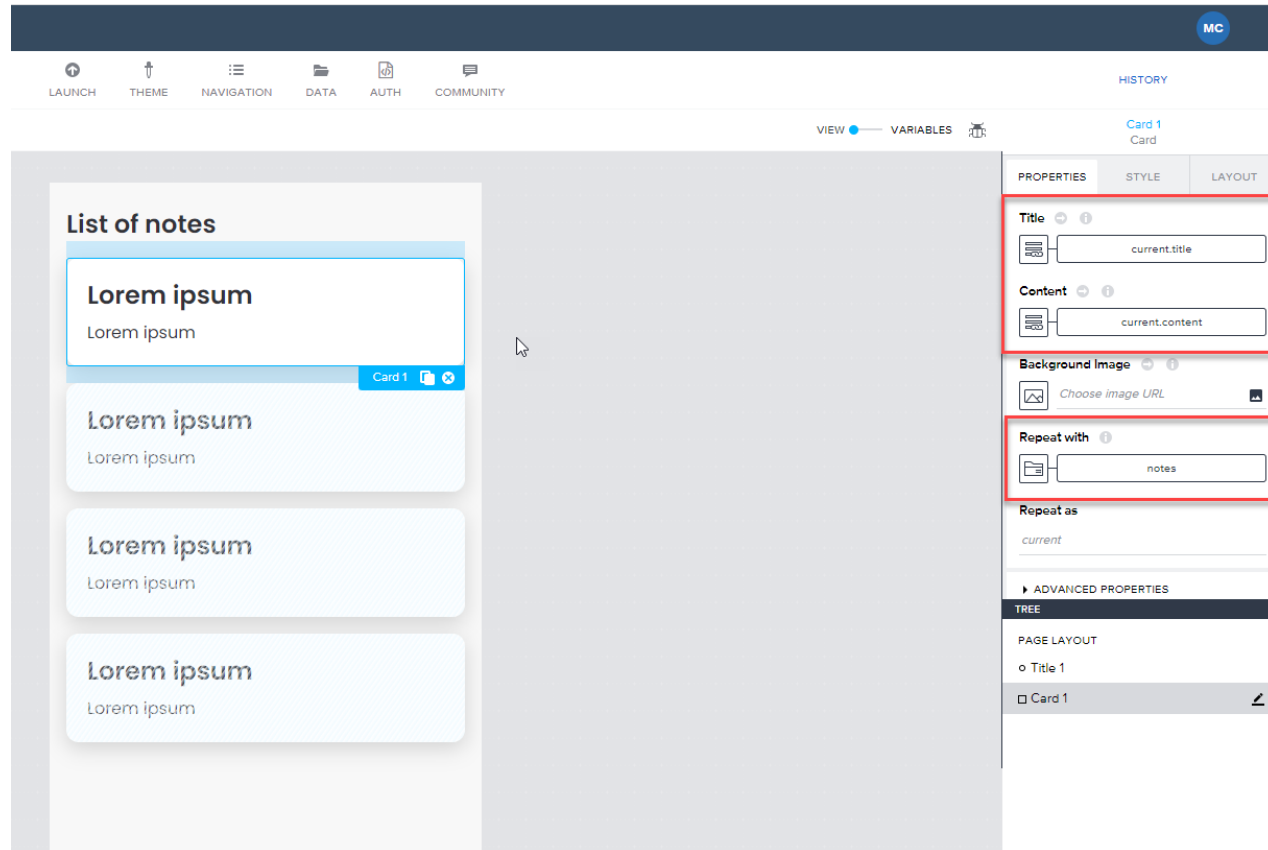


# Quick look at the flow of the application

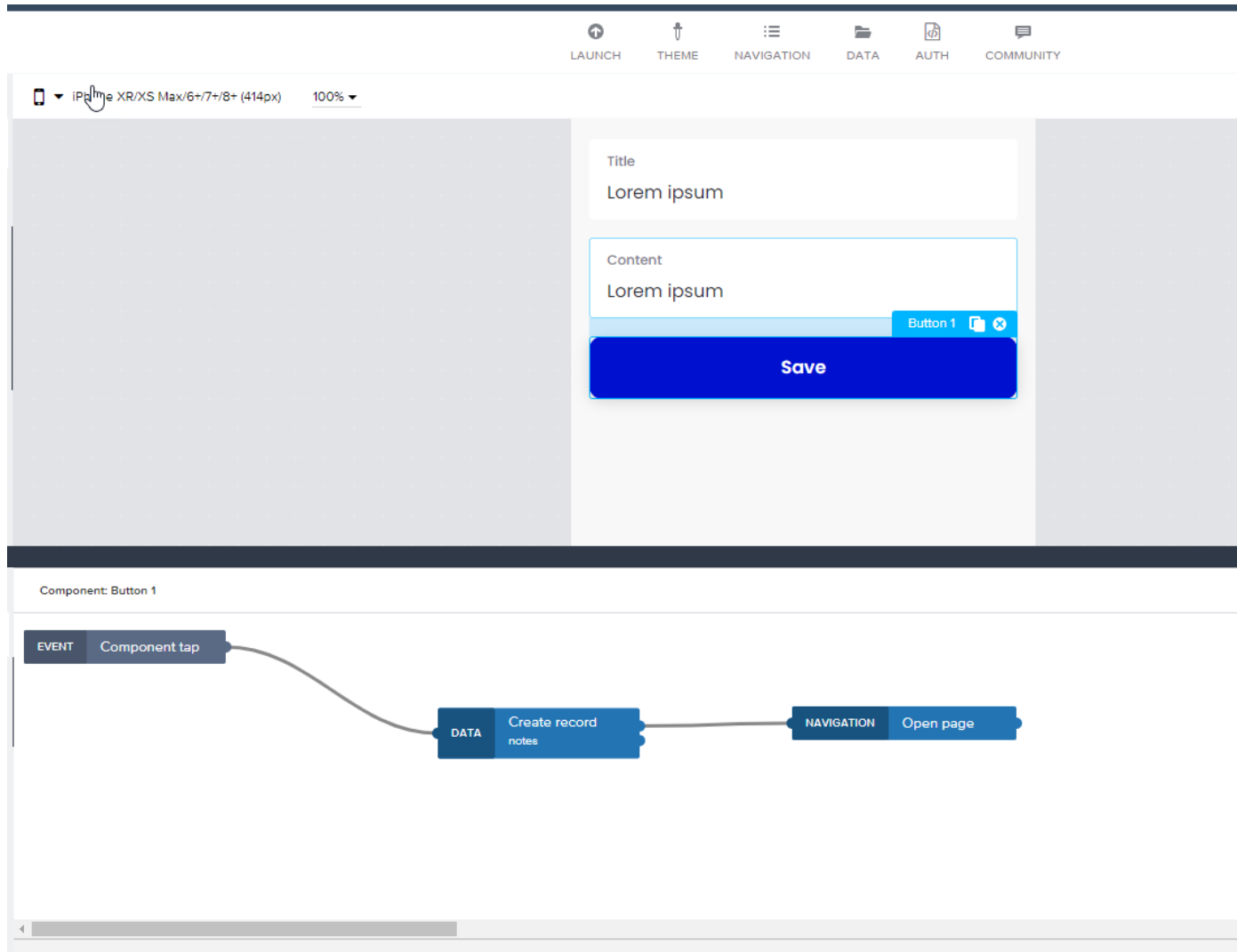


# Step 6 – Bind variables to UI components

Dynamic number of cards... (Needs to be repeated !!)



# Step 7 – Persistence Storage + Navigation



# What have we learned so far

- Introduction to SAP Build Apps
- First mini application with SAP Build Apps
- Building the UI
- Building the logic
- Persisting data
- Binding variables to component

# Key Summary Points – Unit 3

## Pre-requisites

1. Subscribe to [SAP Build Apps](#) via a booster from SAP Business Technology Platform
2. Get the [SAP Build Apps Preview](#) app on your mobile phone



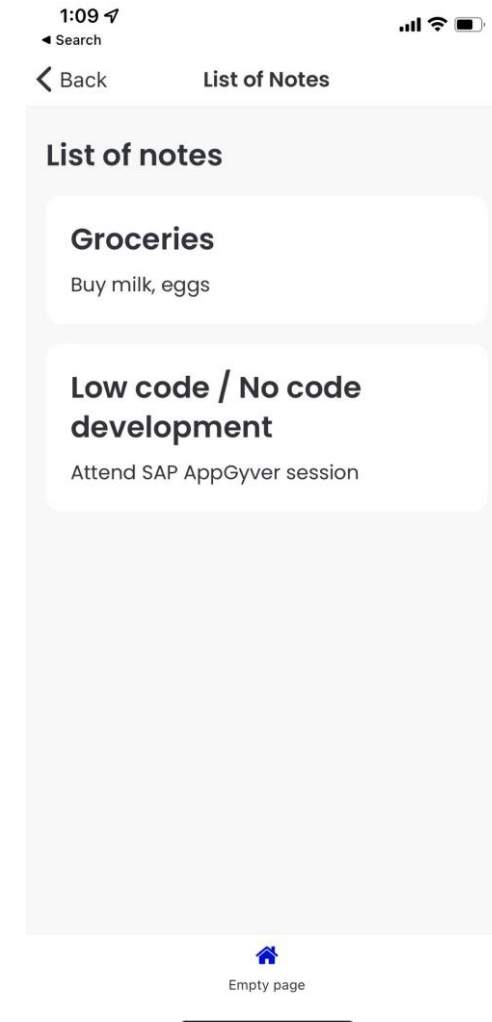
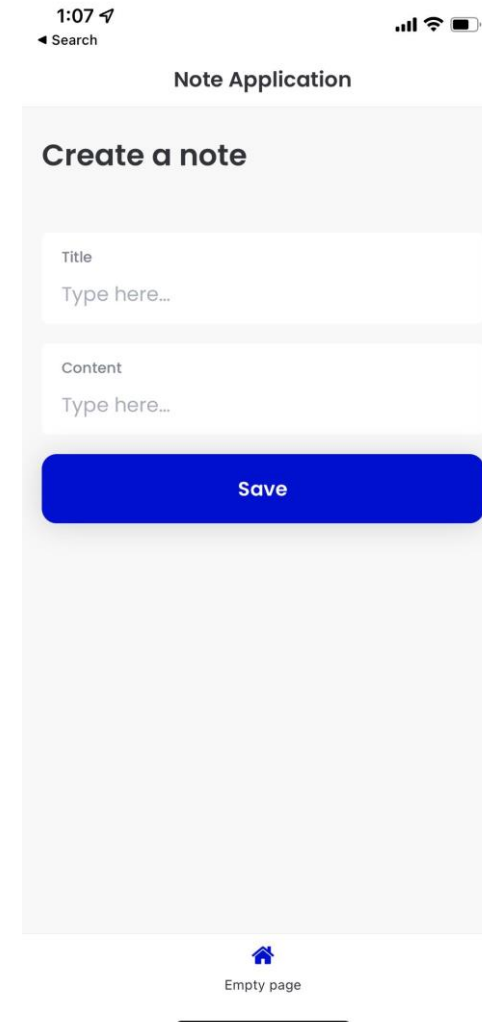
# Key Summary Points – Unit 3

Q8. What are "views" based on, in terms of how they are defined for the user interface of an application?

- ☐ A The data sources for variables.
- ☐ B The flows of logic that need to be performed.
- ☒ C Clear functional distinction, meaning activities that are grouped together.

☒ Correct

Correct. The "views" based on: Clear functional distinction, meaning activities that are grouped together



# Key Summary Points – Unit 3

Q9. What are the three main areas of application development?

☐ A Styling

☒ Data

☒ Logic

☒ User Interface

☒ Correct

Correct. The three main areas of application development are User Interface, Logic and Data

# Key Summary Points – Unit 3

Q10. What are some of the things you did when building your first mini-application in SAP AppGyver?

- ☒ Set up the variables
- ☒ Binding variables to components
- ☐ Implemented authorization capabilities
- ☒ Created the database

☒ Correct

Correct. Here's what you did when you created the first mini-application in SAP AppGyver: Set up the variables, binding variables to components and created the database

# Key Summary Points – Unit 3

## Binding

- Link between variable and component (UI elements)

### Static Binding (Property bound to static string)

Hello world

### Dynamic Binding (Property bound to variable greeting)

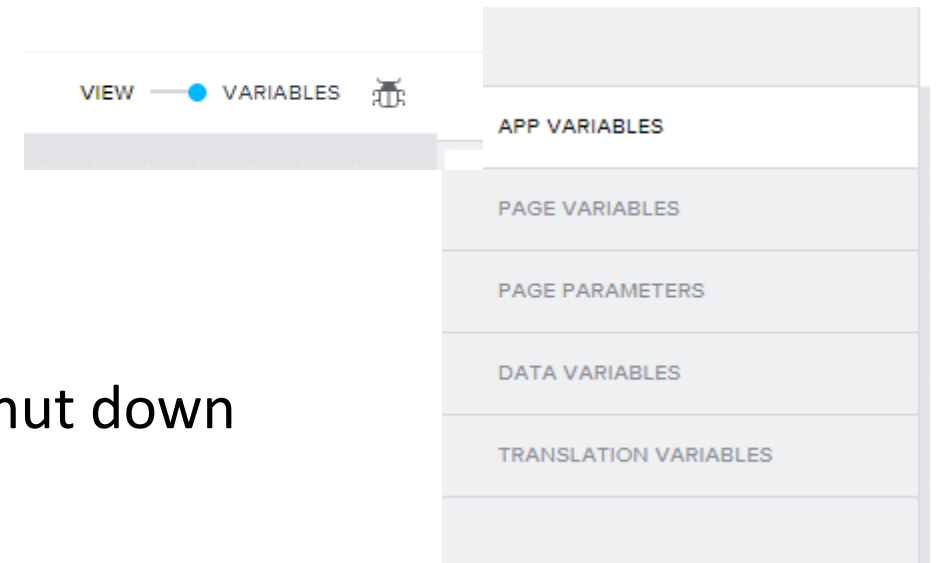
variable greeting = “Good morning !!”

Good morning !!

# Key Summary Points – Unit 3

## Variables

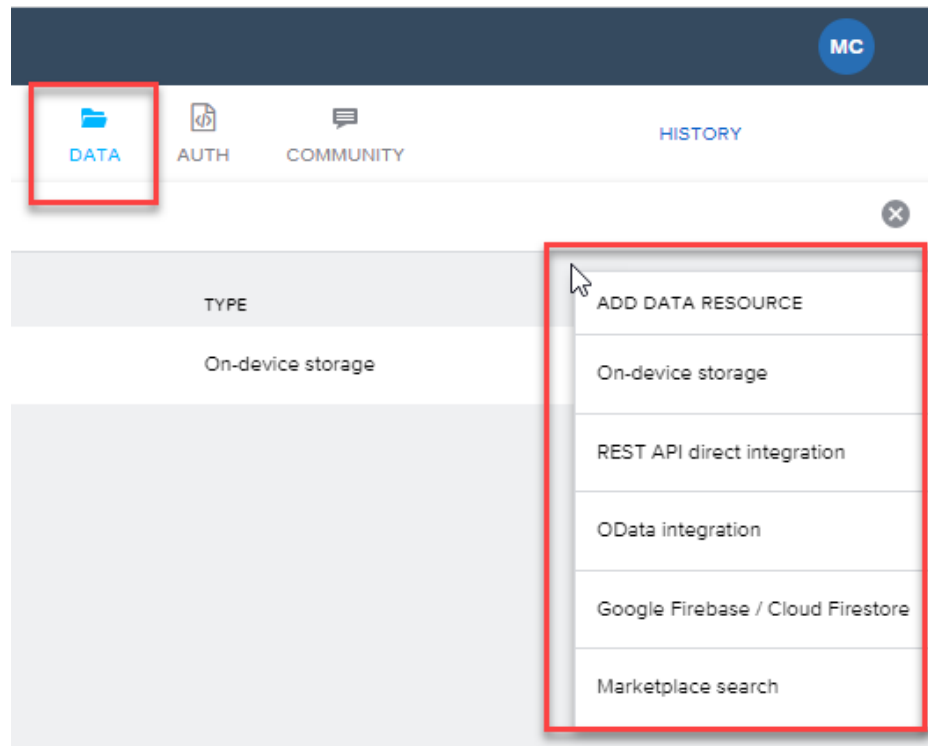
- **App Variables**
  - Exist globally
  - Initialized when app opens, exists until app is shut down
- **Page Variables**
  - Exists in the context of current page
  - Initialized when page opens, exists until page is closed
- **Data Variables**
  - Like page variables, used to read data from [data resources](#)



# Key Summary Points – Unit 3

## Data Resource

- Data that is stored outside the context of the running app
- Data variable is used to retrieve data from data resource



# Key Summary Points – Unit 3

Q2. What is application logic?

- ☐ A Data and metrics integrated to enable performance.
- ☐ B The planning process to develop an application cross-functionally.
- ☒ A flow of events, actions and functions that are driven by variables in data and connected to the UI with bindings.

☒ Correct

Correct. The application logic is: A flow of events, actions and functions that are driven by variables in data and connected to the UI with bindings.