Activities

When you decide to build an App, you have to envisage what you want the App to do.

Envisaging *screens* is probably the easiest way to start, because they are visual.

Mangologic offers a set of screen *types*, or templates, which we call **Activities**.

Look at the screen on the right, for example. In Mangologic terms, this screen is an **Actions Activity** - it gives the user a choice of actions. Exactly which actions is determined by your configuration.
Login Activity

Here is another screen type - another Activity, a Login Activity.

You will likely want your App to start with a Login Activity so that data managed is protected from general access.
### Listing Activity

Another important Activity type is the **Listing Activity**. It makes it easy to get a clickable, scrolling list of things (or people).

For example, if your App manages a service, you might need your users to have a listing of all registered clients of that service.

So you get the idea of Activities now? Think of them as types of screens (there are more types than we have shown so far).

We now turn to an Activity type that is important and unusual - the **Process Activity**.
Process Activity

Look at the Listing Activity on the right. We said it shows clients who have been registered. *How* do they get registered?

A listing, as in the image here, can have a plus sign in the top right-hand of the Action Bar. When it does, then clicking on the plus starts a Process Activity for adding an item to the list.

A Process Activity is an interactive dialogue with the user, where data can be input, and data can be stored in the device’s database.

Processes are also used for other functions besides registering items.
Process Activity

So, unlike other Activity types, a Process Activity is in fact a defined sequence of screens like the one on the right.

The user navigates backwards and forwards through the Process using the arrow buttons.

It is in Processes where you can use complex logical conditions to tailor dialogues to the replies of the user.

You can easily configure complex decision support.
Storyboarding - activities and transitions between them
Let us take a scenario -

Suppose we want to make an App for village health workers.

Let us suppose the idea is to register children in the village in the app, and then we can define a health check process which should be done for each child periodically.

Let us try to “storyboard” this ...
We will now see we can produce this App without doing any programming.
Create Project
When you login you see two tabs: Projects and Deployments.

A **Project** is where we configure Apps and Processes and other things.

A **Deployment** is where we manage the deployment of an App we have created.

Click on the **+Project** button and add a new project for our scenario...
- Give a name for the Project
- Select the default language

Projects can have the status “Test” or “Production”. This is just to help you organise your work. The status of any project can be changed at any time.

- Check the “This is a Test Project” checkbox
- Click Ok button
The new Project appears listed in the Test table.

If you click on the project name, editing functions would appear on the right.

You can “go into” the Project itself by clicking on the Go button.

Click on the Go button and go into the new Project.
The “breadcrumb trail” at the top shows your email and the name of the Project you are working on. Elements of the breadcrumb trail are clickable (for easy navigation). You can click on your email to go back to the top-level.

A default app has been created for this project already, containing initially just a Login Activity.
Now we can return to the scenario. The newly created Project contains just the LOGIN activity.

We will now add the LISTING activity and the PROCESS (registration) activity, and establish the transition between them.
Define registration Process
We are going to keep it simple. We will define a Process with just two components.

First, a *question* - we will ask the user for the name of a child.

Second, an *action* - we will record that name in the database in a new child record.
• Go to the Processes tab and add a Process “Child Registration”

• Go into that Process and add a Text Input interaction (question) to the Child Registration process

• Enter an appropriate question text (e.g. “Enter the name..”)  
• Click Ok button
Now we have a question in our Process. One additional configuration step is needed here.

- **On the left, select the question we created**
- **On the right, click to edit “Field” and enter something meaningful like “kid-name”**

This will allow us to refer to the value entered by the user.
Now to define the *action* to create a new child record in the database.

Databases require defined *structures*.

We need first to define the structure of a child record in order to be able to create one.

In Mangologic, we do that by defining an *Object* type.

If you are familiar with relational databases, to define an Object type is to define a database table.
We will define an Object type to represent the information about a child.

- At the Project level, click on the Objects tab and click "+Object Type"
- Enter "child" as the name of the new object
- Click Ok button

Now we have to add some fields ...
- Go into the child Object
- Select to add a Field
- Give the field the name “name” and the type TEXT
We have defined an Object type “child”.

Now we can complete the registration Process by adding an action which will, each time it is executed, create a new instance (information representing a specific child) of this Object type.

= A new row in the “child” database table.
- Select to add a **Create Object** action to the process.
- Enter a name for the action (e.g. “create child record”)
- Choose “child” as the Object type.
- Click Ok button
We have to configure this action. We are saying create a child record. Now we need to say what information it should contain -

- Select the action on the left
- On the right, select to edit the **Mapping** between fields in the Process and fields in the Object.
On the left you will see listed the fields of the selected Object type (the type is child and we know it has just one field - name).

On the right, we get to choose a field from a question in the Process.

When the action executes and creates a new child record in the database, it takes values from the Process as defined by this Mapping and puts them in the corresponding fields of the object.
Add two Activities to App
Back to the scenario.

Now we can add two Activities to our App - the LISTING activity and the PROCESS (registration) activity.
- Go into the App
- Select to add a LISTING Activity.
- Set the Object Type to be “child”
- Select the “name” field to show in the list for each record (can be configured differently later)
- Click Ok button
- Select to add a PROCESS Activity.
- Select the Process to be Child Registration
- Click Ok button
Define transitions between Activities
A key point to understand.

We have added two new activities LISTING and PROCESS, but they are as yet unlinked.

The only activity actually used in the App remains the LOGIN activity.
This is what it looks like from a storyboard point of view -

The App starts with LOGIN but stays there - no transition has been defined.
Select the LOGIN activity on the left.

On the right you will see the possibility to select an Activity for the “on LOGIN” transition.

Select the “children” LISTING Activity.

Click Ok button.
Here we show this on the storyboard.

We defined the transition between LOGIN and LISTING.

Now we can also link in the registration PROCESS ...
Select the “children” LISTING

A LISTING activity has four possible outgoing transitions.

- Click to select an activity to be the destination of the “on ADD” transition
- Select the registration PROCESS Activity
- Click Ok button
With regard to our scenario, we have now achieved this much. The App configuration contains three links Activities. The Process (registration) contains an action which creates a record in the database.
Test it out
We can test out what we have built by selecting the Test tab indicated by the arrow.

The Emulator which appears does not show you exactly what the App will like on an Android, but it does allow you to test **Process logic** and **App workflow** (transitions).
- Click the **Start** button on the Test tab
  The Emulator will first show a Login button (indicating the Login Activity)
- **Click the Login button**
  The Emulator requires no real authentication - the transition we defined for “on Login” is followed and the Emulator renders the “children” LISTING activity. The listing is of course initially empty.
- **Click the + button as indicated by the arrow to trigger the “on Add” transition and start a registration PROCESS Activity.**
The debugging window now shows the Process structure on the left, and highlights the Emulator’s current position within it.

The question we defined appears.

- **Enter a child’s name in response to the question**
- **Click the forward arrow (top right) to proceed**
The next node in the Process is the action we defined. So that is executed - it creates a database record - and the Process ends because that was the final node.

At the end of a Process, the Emulator shows any database changes which have been made (together with fields and values written to the database. Note: created, modified and uuid are internal fields which exist for all objects).
Now the children list is no longer empty.

The registration Process works as intended and with it, we have created a child record!

This would work exactly the same on an Android.

You have already defined a mini data-collection App! Congratulations!