

# Client: Orange Money, Africa

## Brief: Create a new concept and UX vision for a 'super-app' integrating banking and 'lifestyle' features

Orange Money wanted concepts for a new type of app that would make financial and shopping services available to millions of African customers and become a hub for related shopping and lifestyle activities.

### What I did - main activities

- SD** - Analysed and proposed user research to inform service features and key user journeys (see case study 1).
- IA** - Organised app features and tasks around user goals and mental models, for clear navigation (this case study).
- IxD** - Defined app interactions from an existing pattern library, and defining new patterns (see case study 3).

### Project deliverables

- Wireframes at various levels of fidelity to explore, critique, or define UX thinking.
- Prototypes to explain, validate and test concepts with stakeholders, product team and users.

## IA

### Information Architecture

#### Addressing complexity by organising and naming things meaningfully

Combining existing features and separate products into a clear, simple structure that was easy to navigate and quick to learn meant some methodical analysis and proposing new ways to interact with Orange services.

I proposed a stratified approach where multiple features were grouped in ways that made sense to users, creating clear and simple drill-downs to individual tasks.

### Creating a user-centred view of features and how they are presented and accessed

Senior stakeholders, from a marketing background, were focused on a features-for-features-sake approach. This led to very complicated navigation and low engagement.

The following examples show how I went about achieving simplifying the experience for the user, without losing any of the app's rich functionality.

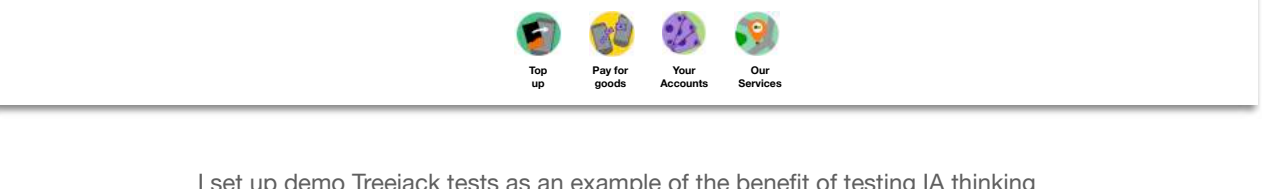
In fact we were able to add even more functions, without making the app feel overwhelming to use.

### Remove repetition, and group features meaningfully

First, I wanted to explain the problem to stakeholders who were quite attached to the old way of seeing the app.

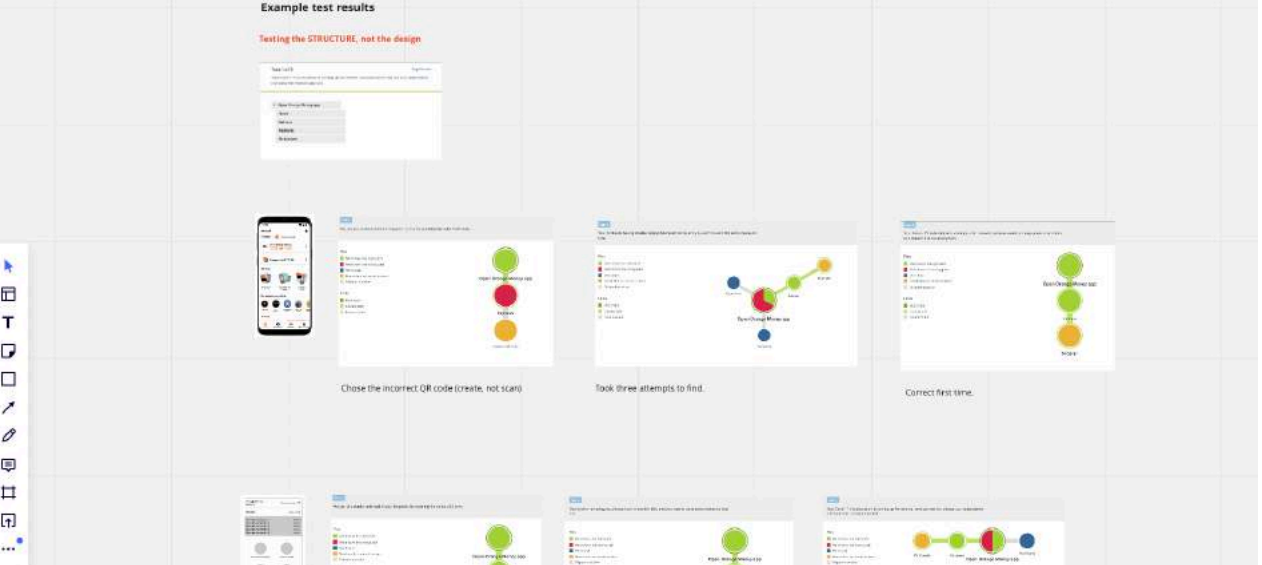
To maintain a positive stance, I proposed solutions to the problems I wanted to illuminate.

Below, I wanted to explain how grouping would simplify things for the user without taking anything away from the app. On the left is the old set of navigation headings and features, with eleven elements; and on the right, the proposed new structure, with four.

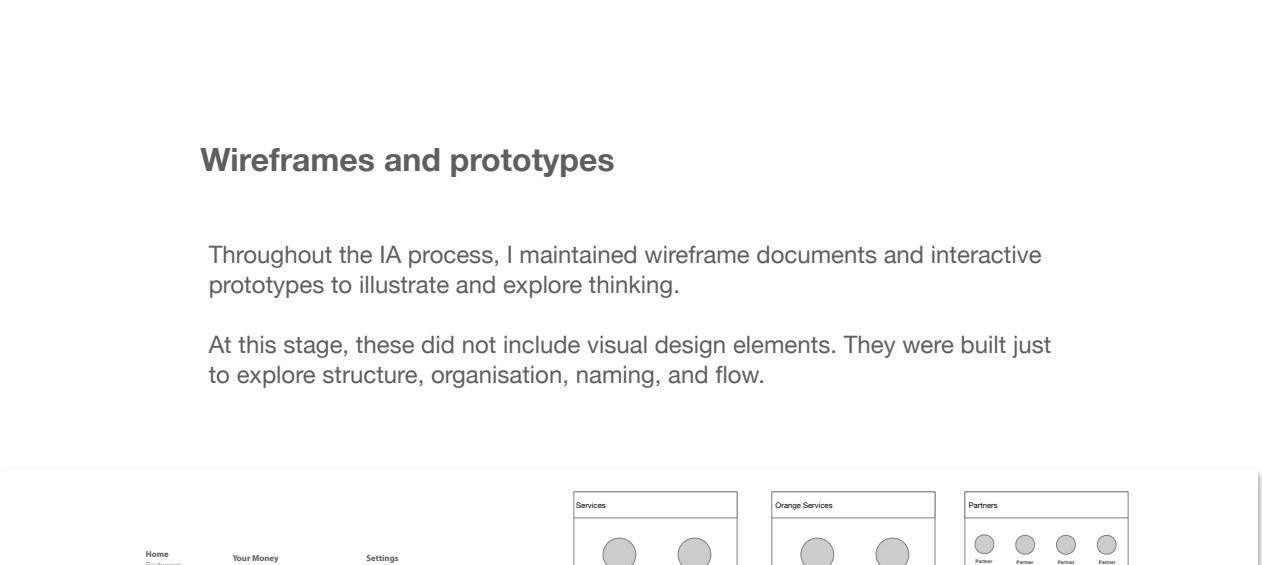


I also showed the impact this would have on the homepage.

For completeness the mini site map showed where the complete set of app features would live and how they would drill down to, following a more user-centred organisation.



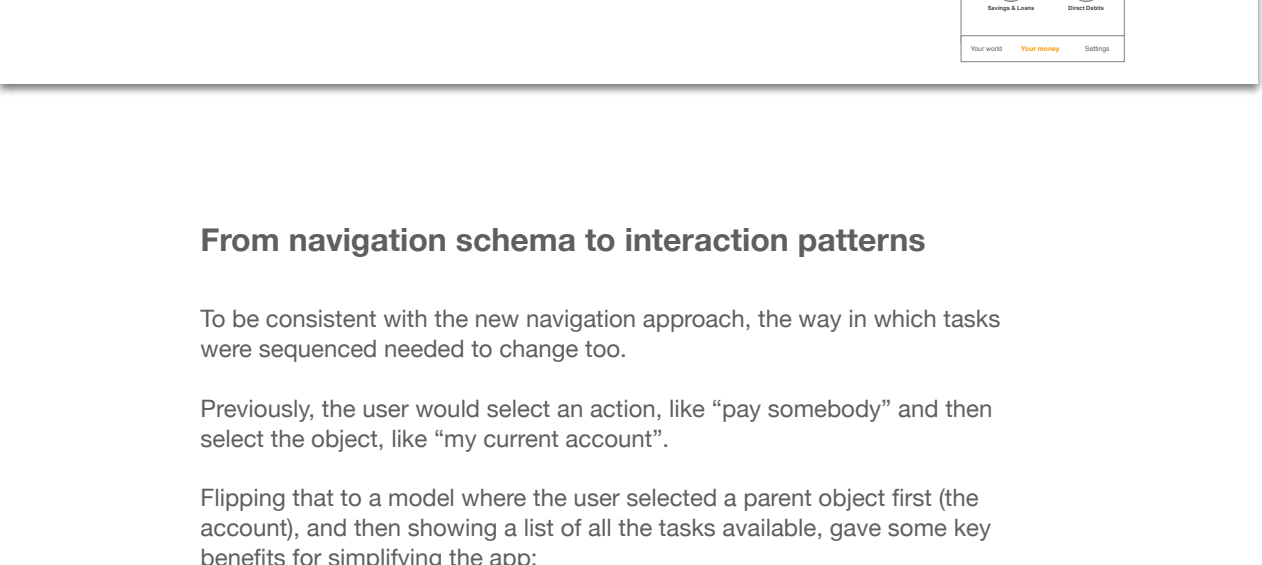
I set up demo Treejack tests as an example of the benefit of testing IA thinking early-on, and to show how much we could learn at relatively little cost. (I shared the results on the team Miro board.)



### Wireframes and prototypes

Throughout the IA process, I maintained wireframe documents and interactive prototypes to illustrate and explore thinking.

At this stage, these did not include visual design elements. They were built just to explore structure, organisation, naming, and flow.



### From navigation schema to interaction patterns

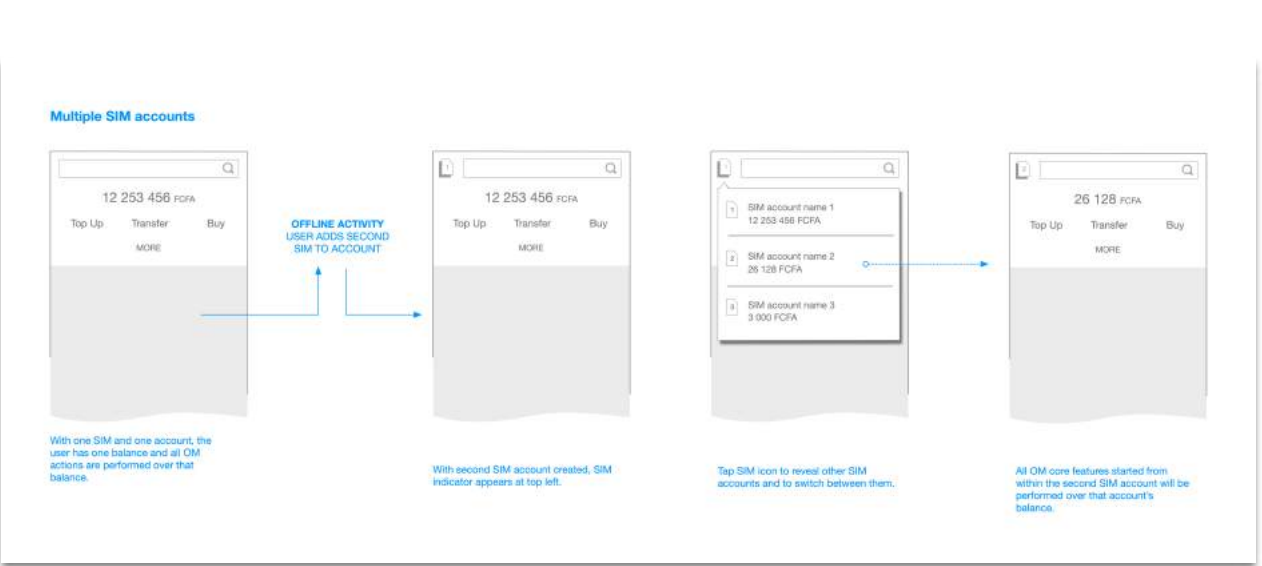
To be consistent with the new navigation approach, the way in which tasks were sequenced needed to change too.

Previously, the user would select an action, like "pay somebody" and then select the object, like "my current account".

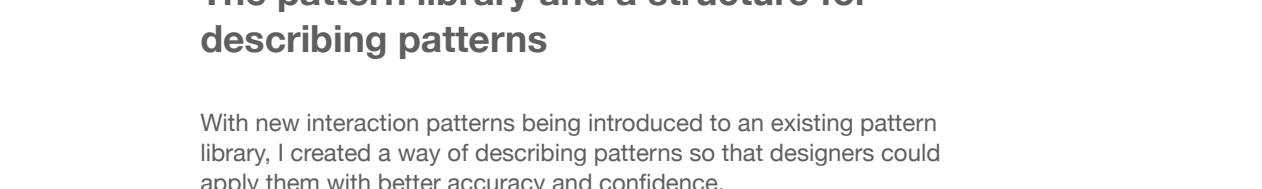
Flipping that to a model where the user selected a parent object first (the account), and then showing a list of all the tasks available, gave some key benefits for simplifying the app:

- Easy to see what you can do at each stage
- You're not bombarded with irrelevant actions
- Less to learn
- Easier to explore
- More task-oriented
- Simpler pages

I created a hypothesis document to explain the problem and its solution, for use in stakeholder discussions and user test planning:



### To bring the thinking to life, I also drew up low-fidelity wireframes to show options for how the thinking could be implemented:



As with the earlier service design phases, I built interactive prototypes wherever necessary to illuminate thinking and to validate, explore or expand upon thinking within the team and with user groups.

### The pattern library and a structure for describing patterns

With new interaction patterns being introduced to an existing pattern library, I created a way of describing patterns, so that designers could apply them with better accuracy and confidence.

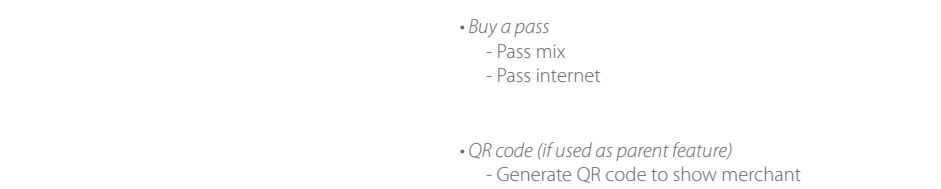
This approach also lent itself to increased governance around the pattern library, which lacked any instructions or directions for applying its various elements.

I proposed a Christopher Alexander model for organising and describing patterns, applying it to the new patterns our team was introducing for illustration.

First, I summarised the approach:

### A pattern for describing patterns

(Drawn from 'A Pattern Language' by Christopher Alexander)



Then I made real-life examples from the work we were doing:

#### Name of pattern: NESTED SELECTION

**Description of pattern**  
If the user needs to select a main, or "parent" feature, then decide between different facets of the feature, or different instances of it, or different ways of using it.

**Related patterns**

- Intelligent defaults
- Tiles
- Tabs

**Orange apps using this pattern**

**Orange Money Anticipation**

- Make a payment
  - Using merchant number
  - With QR code
- Top up
  - This phone number
  - Another phone number
- Buy a pass
  - Pass mix
  - Pass internet
- QR code (if used as parent feature)
  - Generate QR code to show merchant
  - Pass internet
  - Pass internet

**Example application of pattern**

In OMA, to pay a merchant, the user first selects the appropriate 'Pay Merchant' option.

If paying by merchant ID, there are two options to choose from: name or number.



#### Name of pattern: INTELLIGENT DEFAULTS

**Description of pattern**  
Intelligent defaults save the user time and effort by defaulting selections.

The user must be able to change from the default to the other option easily, and without having to start the task again from the home page.

The default selections must match user expectations and might be informed by:

- last-used
- frequently-used
- best option for user's circumstances
- other

**Related patterns**

- Nested selections
- Tiles
- Tabs

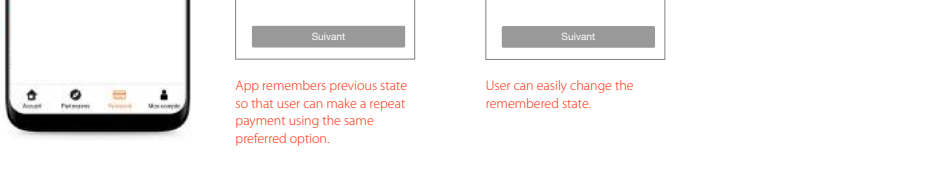
**Orange apps using this pattern**

**Orange Money Anticipation**

- Make a payment
  - Using merchant number
  - With QR code
- Top up
  - This phone number
  - Another phone number
- Buy a pass
  - Pass mix
  - Pass internet
- QR code (if used as parent feature)
  - Generate QR code to show merchant
  - Pass internet
  - Pass internet

**Example application of pattern**

Once a user has paid a merchant in OMA, the app remembers the method chosen and defaults to it next time.

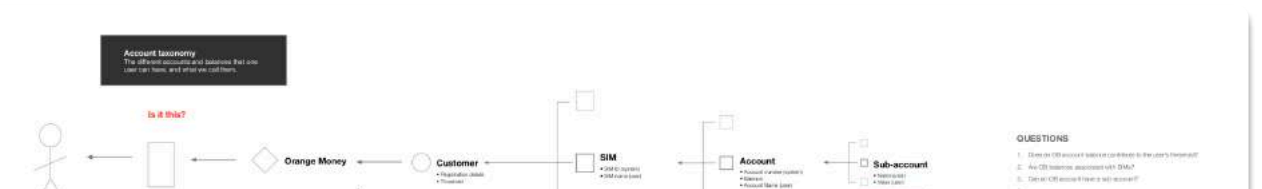


### Taxonomies and vocabulary

With a design team spanning different product groups and management structures, we needed a common vocabulary. Words were being used to describe different things, which created confusion among the team.

I wanted to make sure this confusion wasn't passed on to the user, so set about clarifying a consistent family of names for the things we needed to talk about.

Here, I define the difference between products, bank accounts, SIM-accounts, and 'pots' (sub-accounts):



### Results and outcomes

Some of the work I did in this phase was not expected by the stakeholders, or even other members of the design team. Information architecture is frequently overlooked during the UX process, but I think can make very powerful contributions both to the final product, and to the process involved in getting there.

Thanks for reading.

Steven Hart  
www.hartpartners.co.uk  
stev@hartpartners.co.uk  
07976 628179