

**Peter Sikking**  
man + machine interface works  
Fehrbelliner straÙe 85  
10119 Berlin N-54  
peter@mimiworks.net  
www.mimiworks.net

## Abstract

We present a hands-on user interaction architecture process, that can be adapted to many software project situations, whether desktop, web or mobile.

One of the most important aspects is to define a product vision with the project team, to provide a foundation for further methods.

## Keywords

practical, process, product vision

## 1.0 Introduction

We present a hands-on user interaction architecture process, that can be adapted to many software project situations, whether desktop, web or mobile.

It is the responsibility of the interaction architect to lead the project team to methodically develop the product vision into practical UI solutions that can be implemented by the technical (and graphics) team. So let's get started:

### 2.0 Consensus building

The consensus building phase of the project focuses on raising the awareness in the team how the much the UI is intertwined with the success of the project. There are three phases:

#### 2.1 Product vision

The first thing an interaction architect needs to find out is "what are we trying to achieve here?", and the answer to that is the product vision. If you are uncomfortable with the word product, just substitute project for it.

The product vision defines the purpose of the software under development ("what it is, and for who") and the value it should deliver to the people that pay for it. It needs to be short (aim for three paragraphs) to be useful as a tool during the methods that follow, and

go beyond "deliver functionality x on platform y."

There is no standard template for a product vision; they are as diverse as the teams an interaction architect gets to work with. The best strategy is to coax the team into formulating definitions within their comfort zone, be it functional, marketing, technical or managerial. The interaction architect then helps to refocus the definitions in the needed direction: purpose and value.

By reaching consensus with the team on a product vision, the interaction architect has already delivered a substantial milestone in most software projects. A new sense of purpose has become part of the team.

A couple of direct implications:

- no product vision: no chance of finding good UI solutions. It does not compensate in any way that the team follows the platform's UI styleguide to the letter;
- the product vision forms the basis for taking all further UI decisions in the project. If at a later stage the product vision is changed, all decisions will have to be re-evaluated;
- different teams that work on competing, almost identical software will still have different product vi-

sions. The market demands that for survival. This also means that these teams cannot blindly copy solutions from each other. They will have to work towards the best UI solution from their own product vision.

### 2.2 Technical platform

Although the interaction architect's focus is the final interaction on the screen, it is still important to discuss the technical UI platform with the developers. So that both parties are in consensus how fine-grained event input and screen output can be controlled, and what impact that has on the project.

### 2.3 User scenarios

User scenarios encapsulate the domain knowledge within the team and map out user-space. In order to be useful as a tool during the methods that follow, there need to be few of them (aim for less than ten—yes, it is that tough), and they need to be short, focussing purely on user intent.

Is your number of user scenarios exploding? Start generalising, move up a level. Boiling it down to get to the essence is much more important than being completist.

### 3.0 Evaluation

The goal is to evaluate interaction of the relevant 'raw material' already present, inside the project. This can be:

- the current version of the software under development;
- software prototypes;
- paper prototypes;
- specification documents;
- mock-ups;
- new ideas.

The user scenarios are used here to evaluate the interaction that really matters. The interaction architect gathers qualitative information about how well the user scenarios are supported by the interaction, and in general if the product vision is achieved through the UI.

### 4.0 Analysis

Now all the pressure is on the interaction architect. First the good, the bad and the ugly from the evaluation is analysed. This should already make clear the core issues that need to be dealt with to fulfil the product vision. Further analysis and probing along the user scenarios will then lead towards a solutions model.

### 5.0 Towards practical UI solutions

With the solutions model in place, things are more straightforward from here. There are two distinct paths forward:

1. the interaction architect works (with a team of associates) autonomously on deriving UI construction drawings and additional specifications that then are reviewed with the software development team;
2. more hands-on, the interaction architect sits down with the technical (and graphics) team and leads the derivation of practical UI solutions from the solutions model.

»Es ist erlaubt digitale und Kopien in Papierform des ganzen Papers oder Teilen davon für den persönlichen Gebrauch oder zur Verwendung in Lehrveranstaltungen zu erstellen. Der Verkauf oder gewerbliche Vertrieb ist untersagt. Rückfragen sind zu stellen an den Vorstand des GC UPA e.V. (Postfach 80 06 46, 70506 Stuttgart). Proceedings of the 4th annual GC UPA Track Gelsenkirchen, September 2006 © 2006 German Chapter of the UPA e.V.«



