

## Let's Introduce a Case Study

You've been approached by CPI Corporation, a (fictional) company that builds, markets, sells, and monitors security systems for homes and small businesses. CPI has no Web presence and wants to roll out a "significant" website that will coincide with the introduction of a new line of security sensors and a set of radically new Web-based services. They want your help in the development of the WebApp, which is called **SafeHomeAssured.com**, and at the same time for you to assist them as they create new Web services that will increase their market share.

You have been asked to attend a meeting in which basic ideas are discussed. During the meeting you learn that CPI has engineered a compact, wireless sensor-controller that will become the core element in a new line of commercial and residential security systems that it intends to call *SafeHome*. A snippet of conversation from the meeting is depicted in the sidebar.

### SAFEHOME



#### A Project Begins

**The scene:** Meeting room at CPI Corporation, a (fictional) company that makes consumer products for home and commercial use

**The players:** A senior business manager; a product development manager; a marketing manager; an engineering manager; and you, the Web engineering expert

#### The conversation:

**Business manager (to product manager):**

Okay, what's this I hear about your folks developing a what? A generic universal wireless box?

**Product manager:** It's pretty cool . . . about the size of a small matchbook . . . we can attach it to sensors of all kinds, a digital camera, just about anything using an IEEE wireless protocol. It allows us to access the device without wires. We think it'll lead to a whole new generation of products.

**Business manager (looking at the marketing manager):** You agree?

**Marketing manager:** I do. In fact, with sales as flat as they've been this year, we need something new. We've been doing a little market research, and we think we've got a line of products and services that could be big.

**Business manager:** How big . . . bottom line big?

**Marketing manager:** It's a whole new generation of what we call "home management products." We call 'em *SafeHome*. They use the new wireless interface, provide homeowners or small-business people with a system that's controlled by their PC via the Internet—home security, home surveillance, appliance and device control—you know, turn down the home air conditioner while you're driving home, that sort of thing. We're also thinking about video monitoring and control within a house or business. Just as important, we intend to vertically integrate the product into our monitoring services, allowing customers to access their account via the Web and determine things like when the system is armed or disarmed, what "events" have occurred over a defined time period . . . things like that. We also intend to do most of our maintenance diagnostics via the Web.

**Product manager:** Engineering's done a technical feasibility study of these ideas. They're doable at relatively low cost. Most hardware is off-the-shelf. Software for the Web is an issue, but it's nothing that we can't get done. We already registered a domain . . . **SafeHomeAssured.com**.

[All CPI managers look directly at you and smile.]

**Business manager:** Interesting. Now, I asked about the bottom line.

(continued)

**SAFEHOME (CONTINUED)**

**Marketing manager:** PCs have penetrated a huge percentage of all households in the United States. If we could price this thing right, it could be a killer-App. Nobody else has our wireless box . . . it's proprietary. We'll have a 2-year jump on the competition.

Revenue? Maybe as much as \$30 to \$40 million in the second year.

**Business manager (smiling broadly):** Let's take this to the next level. I'm interested.

And so, a project begins. You'll notice that there are few details at this stage. Many things need to be defined, specified, and then implemented. The internal perception of the product will change, along with the Web-based system that will support it. But that really doesn't matter at this early stage. *SafeHome* has the support of senior management (who see significant profit potential), and you have an opportunity to be one of the team that will get the job done.

We'll return to *SafeHome* and the **SafeHomeAssured.com** WebApp repeatedly throughout this book, using the project as a case study for describing many aspects of Web engineering. But for now, let's return to our introductory discussion of WebApps and examine their similarity to conventional computer software.

**Are WebApps Really Computer Software?**

There's really no debate here—WebApps are computer software in the sense that they are a collection of executable instructions and data that provide both information and functionality for end users. The implication, therefore, is that it's reasonable to expect that we can develop WebApps by heeding some, if not all, of the lessons we've learned during the many decades we've built conventional computer-based systems. It's also reasonable to assume that we'll encounter many, if not all, of the problems (both cultural and technical) that we experienced during the earlier era. But more on all that later in this book.

**Are the Attributes of WebApps Different from the Attributes of Conventional Software?**

There is some debate about the correct answer to this question. Some people argue that a WebApp is nothing more than a client-server application with a heavy emphasis on both aesthetic presentation (e.g., layout, graphics, audio and video elements) and functionality and that both WebApps and conventional client-server applications have the same attributes. But others (including us, the authors of this book) think that when considered in their totality, a complete set of WebApp characteristics do differentiate Web-based systems from more conventional computer-