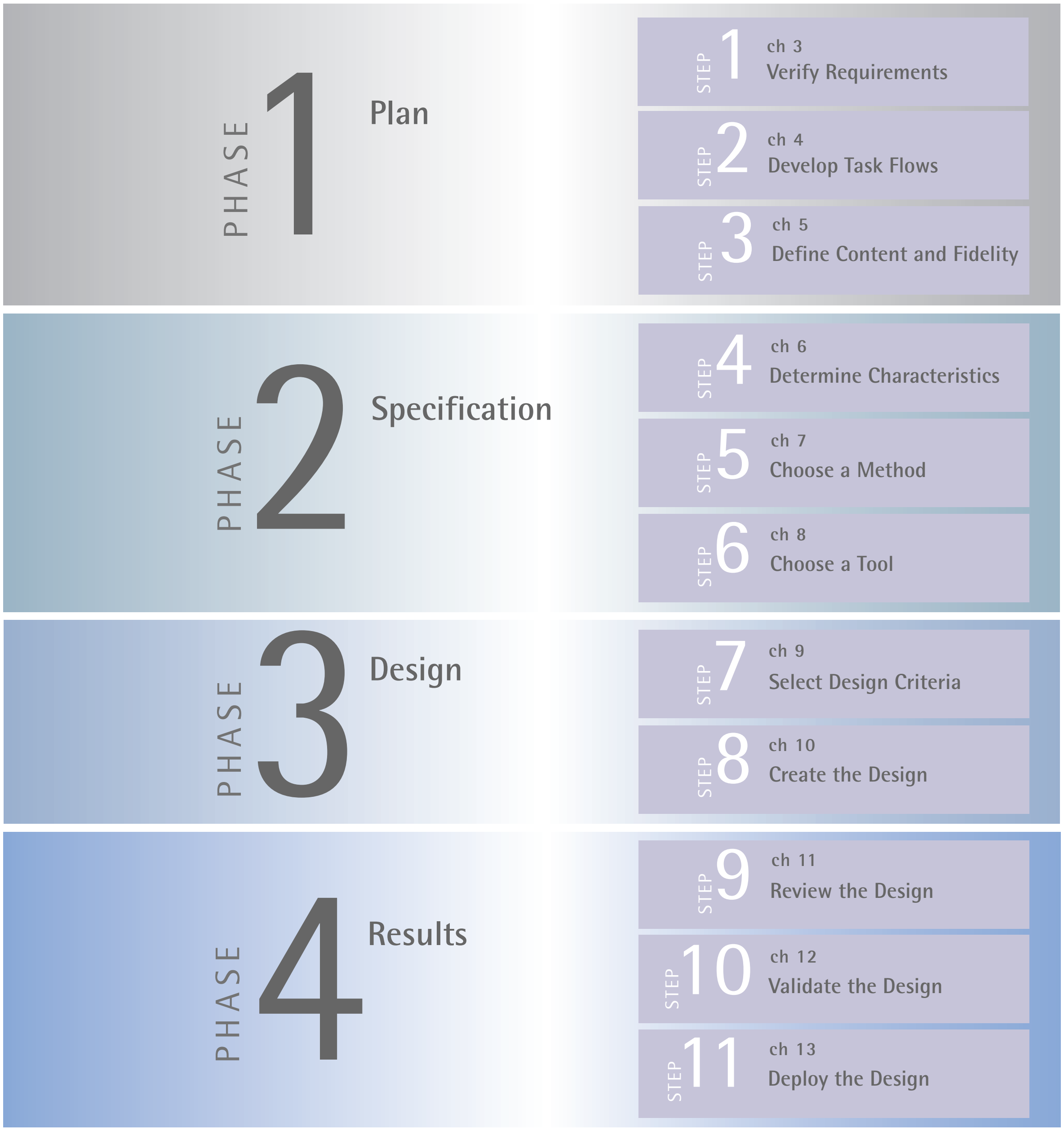
CHAPTER

The Effective Prototyping Process



PHASE 1 Plan

STEP 1 ch 3 Verify Requirements

STEP 2 ch 4 Develop Task Flows

STEP 3 ch 5 Define Content and Fidelity

PHASE 2 Specification

STEP 4 ch 6 Determine Characteristics

STEP 5 ch 7 Choose a Method

STEP 6 ch 8 Choose a Tool

PHASE 3 Design

STEP 7 ch 9 Select Design Criteria

STEP 8 ch 10 Create the Design

PHASE 4 Results

STEP 9 ch 11 Review the Design

STEP 10 ch 12 Validate the Design

STEP 11 ch 13 Deploy the Design

2 THE EFFECTIVE PROTOTYPING PROCESS

Prototyping is a process anyone can learn and master. Although few master every aspect, there is enough latitude in prototyping that we can all find our own niche in using prototyping for communicating software requirements, designs, and ideas. That those ideas may get passed on and iterated, changed, and refined is all part of the process of software creation.

If one chooses wisely and sets expectations accordingly, effective prototyping is a repeatable predictable process of which the results are anything but guesswork. This process is not specialized, complicated, nor lengthy. Effective prototyping is a four-phase process of 11 steps that anyone with an interest can accomplish, whether to create prototypes yourself, manage prototyping in your software creation process, or any other reasonable purpose.

SCALABLE PROTOTYPING STEPS

This book presents a prototyping method in steps that are meant to be scaled to fit the prototyping activity. There will be some steps you will want to follow very closely and use the guided procedures to make sure you get every last detail correct. For quick prototyping, with less of a demand on diligence, you should know the steps and take them into consideration. For example, if you will spend 2 weeks prototyping, then spending half a day planning is not unreasonable. However, if you will just create a quick prototype in an hour, then knowing what the planning issues are and making a more cursory planning would be a reasonable approach.

PHASE I: PLAN (Chapters 3–5)

Phase I helps you to determine prototyping needs and to plan the prototyping process accordingly. You will decide what aspects of the software should or should not be prototyped to provide maximum benefit to your prototyping effort.

Step 1: Verify the Requirements (Chapter 3)

The process starts with determining prototyping requirements. These requirements are not identical to the software requirements but rather are a subset of those based on the audience of the prototype and on your current stage in the software-making process. In determining prototype requirements, you choose a focus for the prototype that influences both the task flow and prototyping content.

Step 2: Create a Task/Screen Flow (Chapter 4)

To effectively prototype, you must have some idea of how the user navigates from one screen/page to the next. Likewise, it is necessary to know what happens when a user clicks on a certain widget (or why s/he would want to). Sketching a task flow is a scalable activity: it can encompass a small or large part of the system or it can involve just one person or the whole team. A task flow can evolve as design and prototypes progress through iterations. Often, it is not simply enough to know the task flow; you must also understand the context in which the task flow takes place. Often, it is necessary to complement the task flow with a scenario or an archetypal story of your task flow.

Step 3: Specifying Content and Fidelity (Chapter 5)

Most prototyping is characterized as either high or low fidelity, with a laundry list of methods or tools thrown into one or the other category. A more comprehensive way to characterize a prototype is by first identifying the prototyping content and then setting that information against a sliding scale of possible fidelity levels. Far from having just one characteristic of fidelity, a prototype can have different fidelity levels for each of the following content types: information design, interaction design and navigation model, visual design, editorial content, branding, and system performance/behavior.

These fidelity levels more accurately characterize the prototype than just high or low fidelity. Several other prototyping characteristics (in addition to fidelity) are covered in the next step.

PHASE II: SPECIFICATION (Chapters 6–8)

The second phase of the prototyping process covers the results of decisions made in the first three steps, in which those decisions allow you to act on the planning phases. Steps 4–6 begin with defining prototype characteristics and end with choosing a prototyping tool.

Step 4: Determine the Right Prototyping Characteristics (Chapter 6)

Failing to use the appropriate prototype characteristics is a major cause of ineffective prototyping. For example, providing your target audience with too many or too few details leads to an ineffective use of your time—either in extra time spent prototyping or time spent on a prototype test that is unable to receive needed feedback. It is important to distinguish between the end users of your software and the stakeholders who will help you make the software. Step 4 details the different characteristics of prototypes: audience, stage, speed, longevity, expression, fidelity, style, and medium.

Next, you map your prototyping characteristics and content to a method that most closely matches your needs.

Step 5: Choose a Prototyping Method (Chapter 7)

Step 5 discusses how to decide which method is right for your current situation. A helpful table and worksheet are included at the end of the chapter to assist you in choosing the right method.

A WORD ABOUT OUR WORKSHEETS

The worksheets in this book are works in progress and will be subject to revision and specialization as people outside of the domains we are familiar with begin to use them. You should view the worksheets in this book as a current snapshot of possible future iterations that we plan to make available on our web site: www.effectiveprototyping.com For free updates to these worksheets as well as ability to participate in any discussions about them we invite you to our website.

Step 6: Choose a Prototyping Tool (Chapter 8)

Step 6 matches your prototyping tool to the method you selected in Step 5. We encourage you to prototype with anything you desire because we believe it is more empowering to use a skill set you already possess and a tool you're already familiar with. You can maximize the creative time spent prototyping rather than succumbing to the steep learning curve of a less familiar prototyping tool. Chapters 22 through 26 cover some prototyping tools you may already have on your computer but perhaps never realized their uses for prototyping: Office productivity applications (Microsoft Word, Excel, and PowerPoint), Visio, and Acrobat.

PHASE III: DESIGN (Chapters 9 and 10)

After specifying a prototyping strategy, Phase III focuses on executing the prototype through good design. For the accomplished prototyper, good design is already part of the professional practice, and these steps may seem naive or too simplified. For the first-time prototyper, Phase III provides a handy guide to learning how to prototype, including basic steps, good practice, and a guide through making design choices and how to execute them.

Step 7: Formulate Design Criteria (Chapter 9)

A key factor in effective prototyping is the ability to defend your prototype's design. Savvy designers have a unique sensibility for combining design guidelines with technical, end-user, and business requirements to form an elegant solution. A design rationale, based partly on design guidelines, is one of the more successful methods for outlining design decisions. However, not everyone is a designer, and it is not necessary to become one. Step 7 lists some best practices in design guidelines from the fields of cognitive psychology, graphic design, and information design. These design guidelines ensure that your page/screen compositions are not arbitrarily conceived. Even if your prototype cannot be considered the best design, it should be understandable to a general audience, and you should be able to explain the rationale for your decisions.

Step 8: Create the Prototype (Chapter 10)

Step 8 discusses methods for tying together guidelines and requirements to achieve best practice design. In the end, the quality of your prototype is based on the quality of user research, accurate definition of requirements, and your own design exploration/iteration and analysis. Your analysis can only be as thorough as your own well-rounded understanding of the guidelines and requirements as well as an appreciation for the needs of your audience.

PHASE IV: RESULTS (Chapters 11–13)

Of all the activities of prototyping, the results phase is the one that is perhaps most thoroughly covered by other texts. Therefore it is also where we spend the least amount of our attention. This section is more for the novice who:

- Is not familiar with the proper way to conduct prototype reviews
- Has never been involved with the activities and issues of usability testing
- Has little experience creating a prototype and converting it into a product

These chapters are mostly high-level discussions with pointers to where you can find more information on these well-covered topics.

Step 9: Review the Prototype (Chapter 11)

Step 9 outlines reviews with internal stakeholders and ways to ensure that an effective prototype goes on for validation. Likewise, this chapter discusses the issues around reviews: what to look for and what strategies to use.

Step 10: Validate the Design (Chapter 12)

Step 10 discusses prototype validation through usability testing and other validation techniques with external stakeholders.

Step 11: Implement the Design (Chapter 13)

The last step in prototyping is taking an iterated prototype and shaping it into a product or service concept as part of a new technology incubation process or translating it into an actual product or service to deploy to the marketplace. Implementation involves the actions required to realize a prototype appropriate to the goals and objectives of the creators.

SUMMARY

The prototyping process we described follows four phases: plan, specification, design, and results. Each one of these phases has multiple steps which are shortly described here and will be covered by the following chapters in this book.

ARNOSOFT CONTEMPLATES A PROTOTYPE

The project kick-off day has arrived. The company is reviewing its development methodology, a variation on the waterfall process, which involves some very specific engineering-driven steps. First, market requirements are drawn up. The engineers create a functional design document. The engineers develop a design. The design team makes it look good, and then the system is coded. Over the course of the past year, some elements of user-centered design have been introduced, such as some limited user research after the market requirements are done and usability testing during the quality assurance system test (which Ina likened to closing the barn door after the horse has escaped).

The first step in their process is a discussion of the marketing requirements. The ArnoSoft team is busy brainstorming requirements when Reed Dish from Acme Ceramics shows up.

Reed looks at the project schedule they are drawing on the board. He is upset that it will take so long to see an actual product. He wants the process to go much faster.

"I want to see what the thing will look like now. I don't want to sink a ton a dough into something I don't like. Make something, make something now you can . . . you know click through, do the click-click thing. Then I can decide whether it's worth my time or not."

Ina knows it's too early for design. "We can't come up with a design now; it's too soon to make the product. Or do you mean a prototype, a throw-away prototype?"

"Yeah, that's it, a prototype, but make the thing, you know, click so I can see how it works."

"A working prototype?" asked Dirk Spine, "That's extra work from my team. Our deadlines will slip. We can't possibly do that. Full stop."

"Oh come on, we need a 'can do' attitude around here!" said Reed. "I like this idea of a working prototype."

"I said a throw-away prototype," replied Ina.

"Whatever, just make it click. Oh I have to go, I got another sales call to make."

"But we're just getting started here," said Dirk.

"Well just see to it we get a click thing prototype, I gotta go pay for your salaries, later dudes." And with that Reed was gone.

"The click thing?" asked Art.

"So what do we do now?" asked Ina.

"No prototype, full stop," said Dirk.

"Maybe I can put something together," said Ina.

"How?" asked Dirk.

"I can just sketch some wireframes real quick," said Art.

"They won't click," brooded Dirk, "Whatever that means."

"Well, let's think this thing out, what would be the most effective way to prototype?" asked Ina. Let's see what Alfredo thinks.