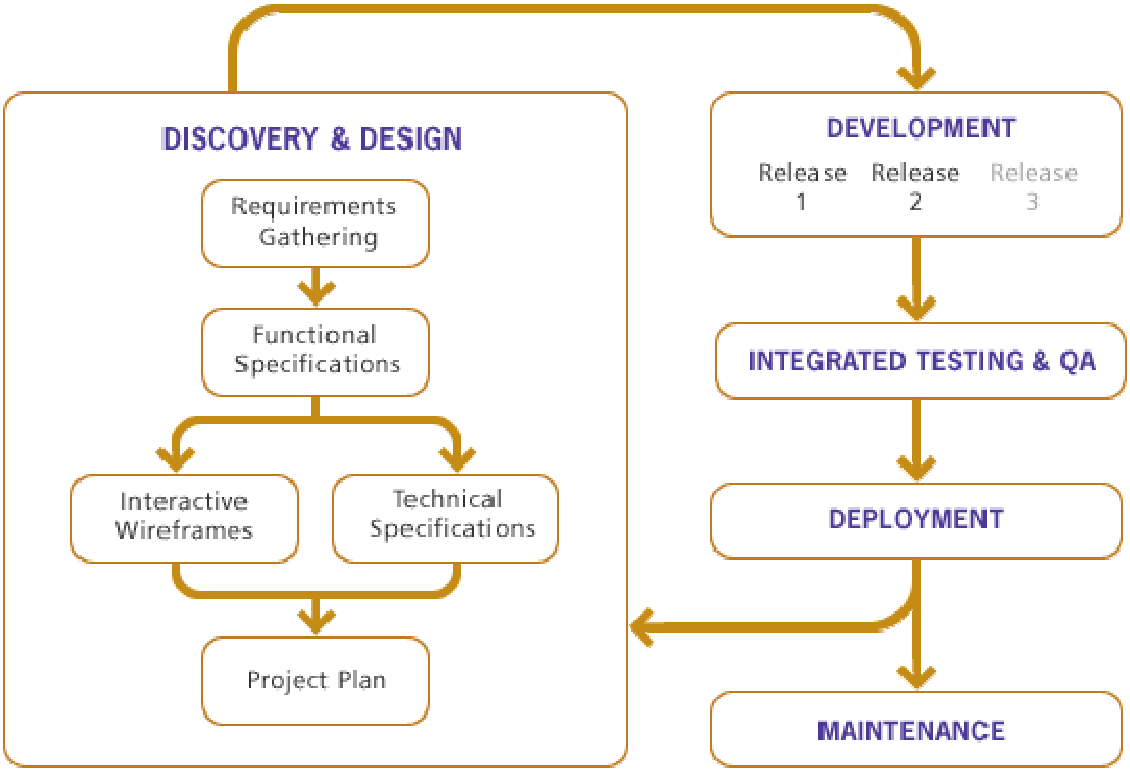
The Bootstrap Process

Table Of Contents

I. Overview

At Bootstrap, we understand the importance of process and proper documentation. Over the past five years we have developed a process that is simple, efficient and has proved successful again and again on a variety of projects. Proper documentation ensures that all parties are "on the same page" and there are no losses in efficiency due to misunderstandings. In fact, good planning, although it takes some time up-front, can end up saving a tremendous amount of time during development. We also know that thorough documentation will extend the life of any system and allow its future development, maintenance, and enhancement to be as painless as possible.



As the diagram above illustrates, the overall process is cyclical and we acknowledge that every application is a living, breathing entity that evolves organically over time as businesses grow, technologies change, processes mutate and lessons are learned.

Discovery & Design is billed on an hourly basis, at the end of which every party knows, in great detail, what will actually be built and we can generate a fixed fee for the remaining Development and Testing. Our maintenance agreements are generally a fixed monthly fee.

Our process is, of course, not set in stone and can be customized for each project. We also continue to refine it with each project and as technologies and tools evolve.

II. Discovery & Design

A. Strategy

At the outset of any project, it is important to define the goals of the project and develop a general strategy to achieve those goals that can be the guiding set of principles throughout the execution of the project. We work with clients to help define, in very clear terms, what the goals of a given project are and lay out a plan to achieve those goals. This usually takes the form of a brief (one to two pages) document that summarizes these goals and strategy. In the case of most development projects, it may also bullet-point general functionality and may define multiple development phases and outline general functionality to be included in the various phases.

B. Requirements Gathering

Once the strategy and goals have been defined, we go through a series of interactive (ideally in-person) whiteboard sessions where, among other things, we discuss:

- Users of the system
- Functional requirements
- Key data points
- Functional flows

We dive deeper into the more esoteric or highly complex areas of the system. It is during this process that we absorb detailed domain knowledge and subtleties of business operations and structures. It is also during this phase that we discuss business process (re)engineering to leverage available technology and create efficiencies.

C. Functional Specifications

During and after the Requirements Gathering phase, the Bootstrap team will be creating a Functional Specifications document. This document provides a foundation for both the graphic design team and the programming and database team. It is intended to outline the scope of the initial rollout. The information in the Functional Specifications document is generally presented in a user-centric fashion. All system users and their individual functions are defined.

That is to say, each system function is described in the context of the user who executes that function.

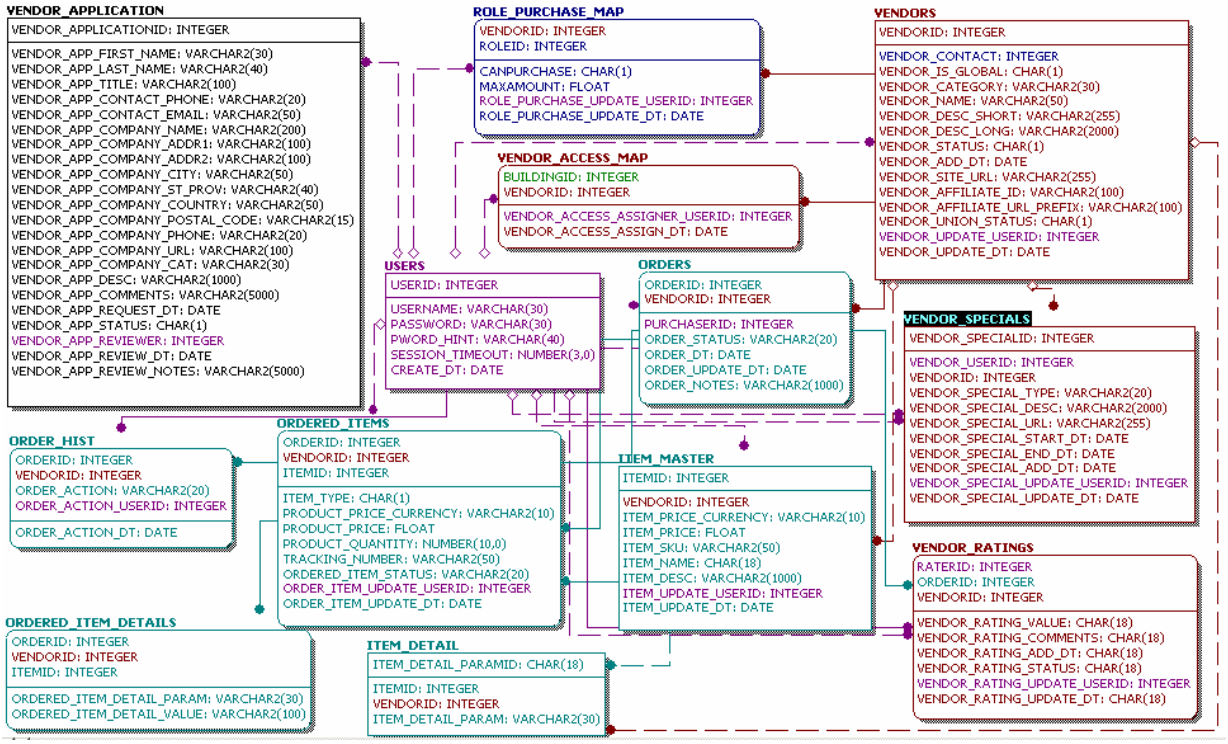
Completely automated tasks are described as well, so that a complete picture of the entire system can be conveyed to anyone who understands the client's business regardless of his or her level of technology expertise. This document is versioned a number of times as clients review drafts and provide feedback and the exact functionality begins to crystallize. Before moving forward, all parties must agree that the functionality outlined in this document represents everything that will be built during the development cycle.

To use the analogy of building construction, the Functional Specifications details what the people working or living in a space will do; in a very precise and comprehensive manner.

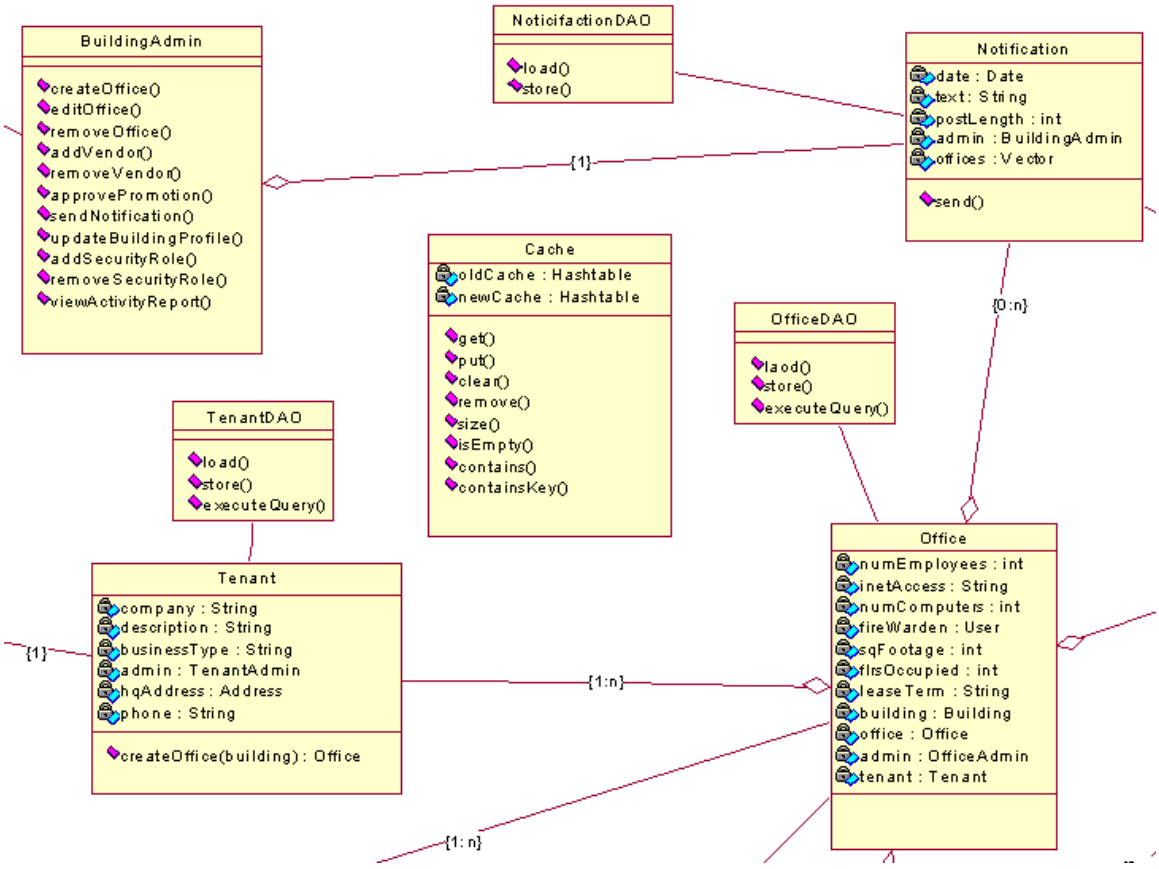
D. Technical Specifications

From the final Functional Specifications document, a Technical Specifications document is then produced. This document includes (as needed) detailed application architecture, data model, object model, hardware/software/bandwidth recommendations, and/or network topology diagram.

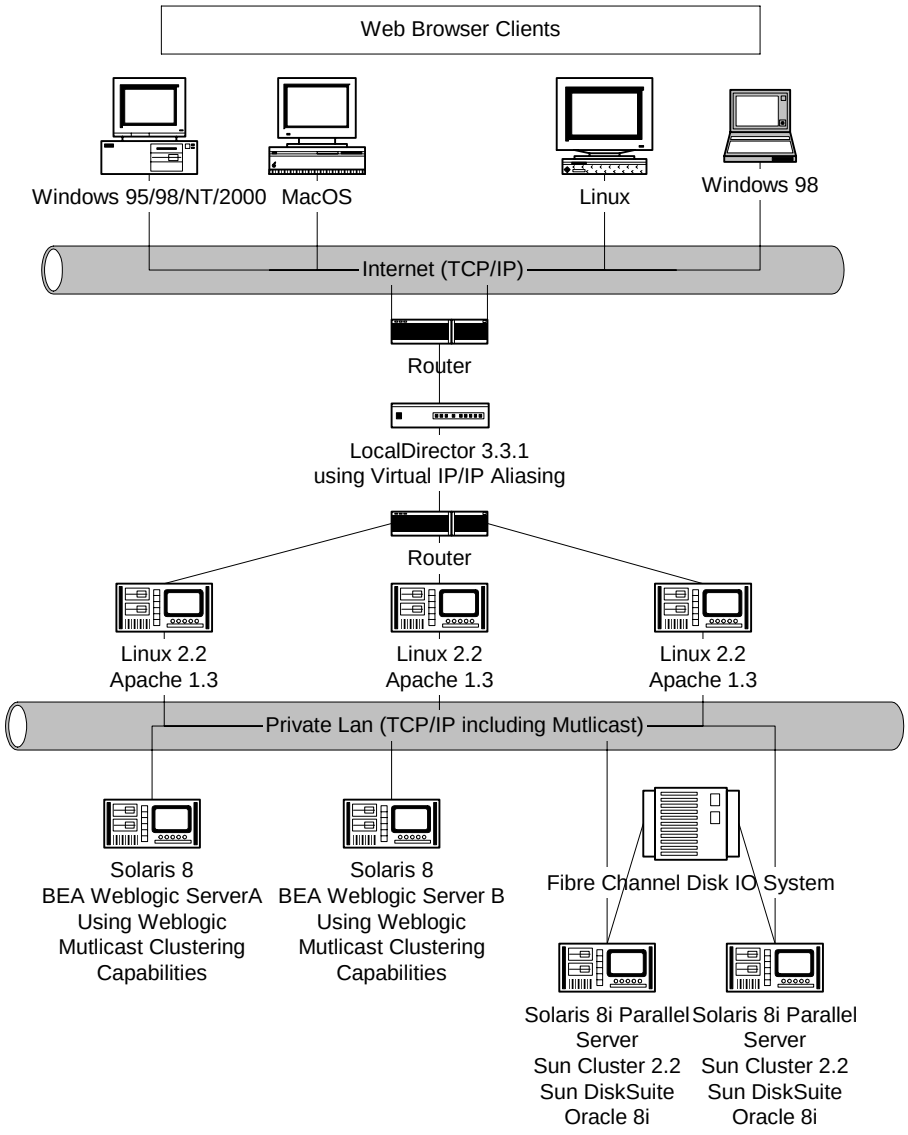
The data model includes all of the tables in the database, the fields defined therein, as well as relationships between the tables. The data model also contains descriptions of each of the data fields in terms of business rules (if applicable). A portion of a Data Model for a relational database can been seen below:



The object model depicts the classes that will make up the software architecture of the middleware server. The middleware represents the business logic of an application. Classes are the building blocks used in object-oriented programming, a programming design approach that allows for easy maintainability, code reuse, and rapid development. A portion of an object model can be seen below:



A typical network topology diagram shows all of the hardware and 3rd components of the system and how they interact. An example is shown below:

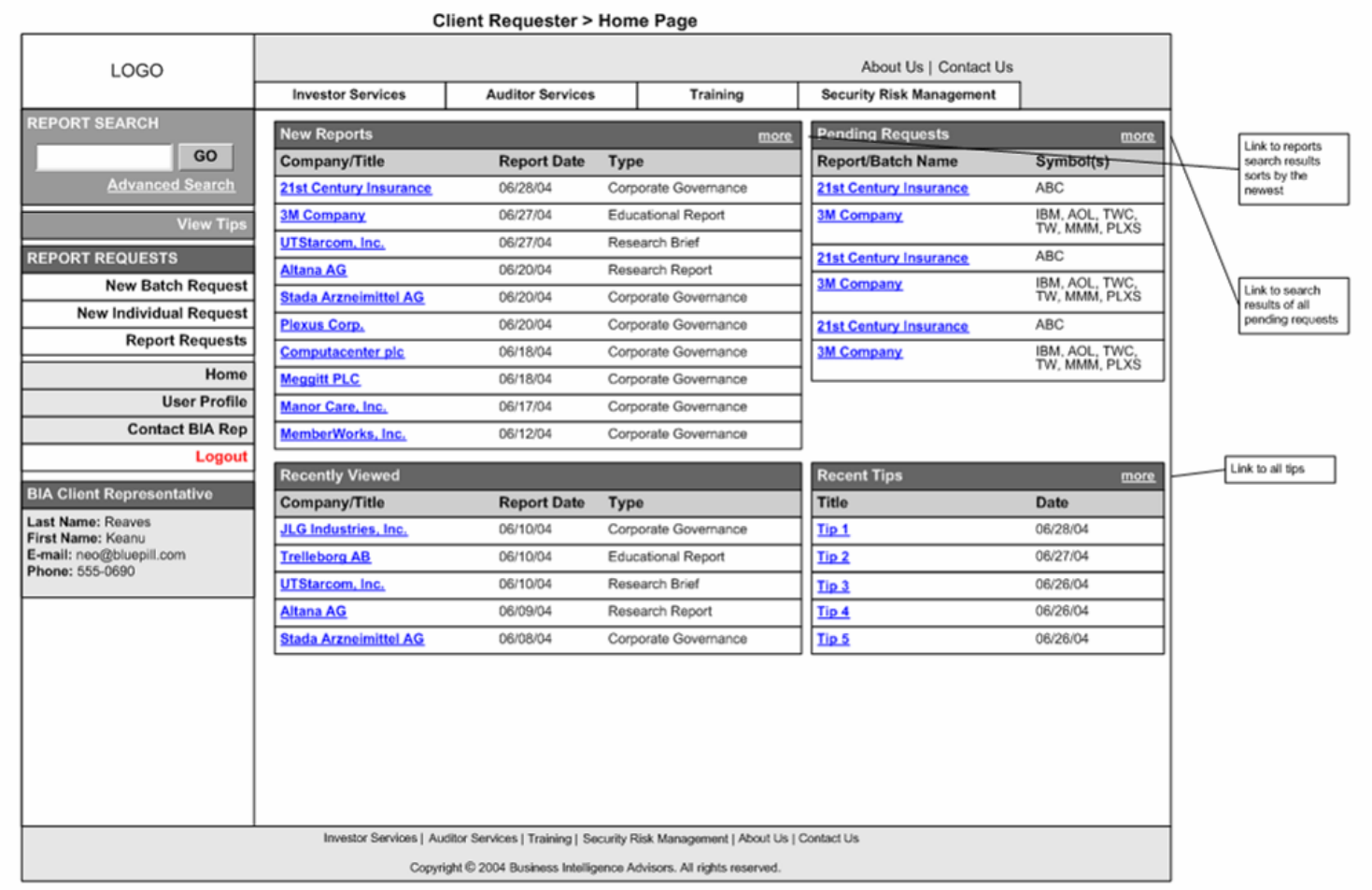


The Technical Specifications serve as a blueprint for the database and programming teams in bringing the back-end of the site into a fully functioning, extensible and maintainable existence. As when constructing a building, one must begin with a well-planned, quality-assured and efficiently designed blueprint in order to produce a quality, highly functional application. The Technical Specifications represents that strong foundation.

E. Interactive Wireframes

Once the Functional Specifications are complete, the wireframe process begins in parallel with the Technical Specifications. The goal of wireframes is to define the interface of the application in great detail. While they are not truly "functional" in any sense and no data is actually retrieved or stored, we create clickable html wireframes that give the reviewer a good sense of the flow from screen to screen. It is usually at this point that most clients begin to really feel and understand what the living application will actually be like to use. As with most parts of our process, the wireframes go through iterations and may be versioned several times until the group is certain that the architecture of the interface is correct.

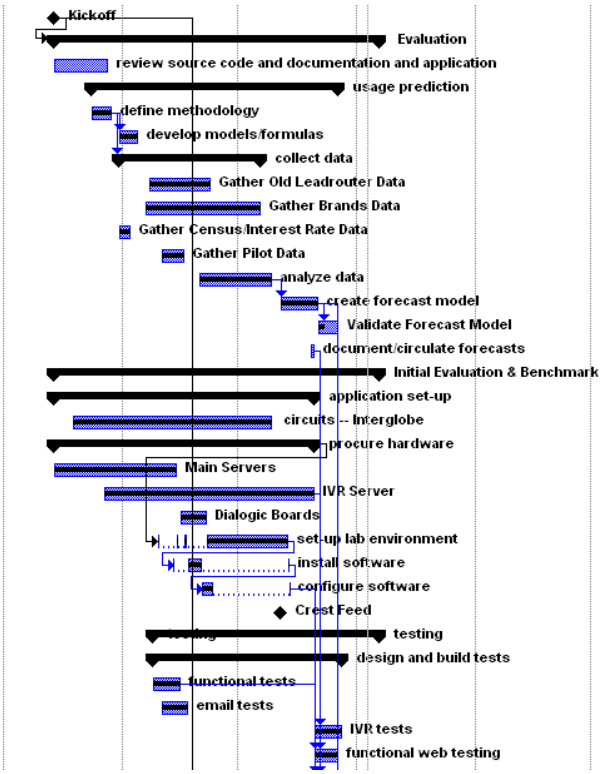
No graphic design is applied at this point (although it can be created in parallel development). The screens do, however contain all links, dynamic data elements, and form elements. They may also contain "call-outs" to specific areas of action or data. An example of a wireframe can be seen below:



F. Project Plan

As a final part of the Design & Discovery phase we create a detailed project plan using Microsoft Project that includes all tasks, phases, resource assignments, dependencies and milestones. An example of a part of a plan can be seen below:

		Task Name	Duration	Start	Finish	Predecessors
2	✓	Kickoff	0 days	Fri 5/14/04	Fri 5/14/04	
3		**Evaluation**	**60 days**	**Fri 5/14/04**	**Fri 8/6/04**	2
4		review source code and documentation anc	10 days	Fri 5/14/04	Thu 5/27/04	
5		**usage prediction**	**45 days**	**Mon 5/24/04**	**Mon 7/26/04**	
6	✓	define methodology	5 days	Mon 5/24/04	Fri 5/28/04	
7	✓	develop models/formulas	5 days	Mon 5/31/04	Fri 6/4/04	6
8	✓	**collect data**	**26 days**	**Mon 5/31/04**	**Tue 7/6/04**	6
9	✓	Gather Old Leadrouter Data	12 days	Tue 6/8/04	Wed 6/23/04	
10	✓	Gather Brands Data	21 days	Mon 6/7/04	Tue 7/6/04	
11	✓	Gather Census/Interest Rate Data	3 days	Mon 5/31/04	Wed 6/2/04	
12	✓	Gather Pilot Data	4 days	Fri 6/11/04	Wed 6/16/04	
13	✓	analyze data	14 days	Mon 6/21/04	Fri 7/9/04	
14	✓	create forecast model	8 days	Mon 7/12/04	Wed 7/21/04	13
15		Validate Forecast Model	3 days	Thu 7/22/04	Mon 7/26/04	14
16		document/circulate forecasts	1 day	Tue 7/20/04	Tue 7/20/04	
17		**Initial Evaluation & Benchmark**	**60 days**	**Fri 5/14/04**	**Fri 8/6/04**	
18		**application set-up**	**47 days**	**Fri 5/14/04**	**Tue 7/20/04**	
19	✓	circuits -- Interglobe	37 days	Wed 5/19/04	Fri 7/9/04	
20	✓	**procure hardware**	**47 days**	**Fri 5/14/04**	**Tue 7/20/04**	
21	✓	Main Servers	22 days	Fri 5/14/04	Mon 6/14/04	
22	✓	IVR Server	38 days	Thu 5/27/04	Tue 7/20/04	
23	✓	Dialogic Boards	5 days	Wed 6/16/04	Tue 6/22/04	
24	✓	set-up lab environment	15.8 days	Thu 6/10/04	Tue 7/13/04	20
25	✓	install software	2 days	Mon 6/14/04	Wed 7/14/04	24
26	✓	configure software	3 days	Mon 6/21/04	Wed 7/14/04	25
27		Crest Feed	0 days	Mon 7/12/04	Mon 7/12/04	
28		**testing**	**42 days**	**Wed 6/9/04**	**Fri 8/6/04**	
29	✓	**design and build tests**	**34 days**	**Wed 6/9/04**	**Tue 7/27/04**	
30	✓	functional tests	5 days	Wed 6/9/04	Tue 6/15/04	
31	✓	email tests	5 days	Fri 6/11/04	Thu 6/17/04	
32	✓	IVR tests	5 days	Wed 7/21/04	Tue 7/27/04	22
33	✓	functional web testing	4 days	Wed 7/21/04	Mon 7/26/04	30,16



III. Development

Once we have finalized the project schedule, development can begin in earnest. The client will receive regular updates from the Technical Lead and Project Manager and the frequency of these meetings or phone calls will be adjusted to fit the needs of the project. Typically, we encourage clients to set aside weekly or bi-weekly appointments to receive updates. These sessions are critical to the project in that they allow the development team to gather the client's feedback on the work completed to date.

As certain project milestones are achieved, the client will receive delivery of any stand-alone components completed to date. The client's feedback permits the development team to make any necessary real-time adjustments to the project, ensuring that the final product meets the client's expectations.

A. Look and Feel

The overall look and feel of the application is developed by our creative staff based on a combination of client input, existing design assets (logo, print collateral, other applications), and fresh thinking. We have found that developing an overall look and feel for a web application is generally an iterative process that works best when clients take an active role. This step determines more than simply what colors and fonts will be used on the site -- the outcome helps determine the ethos of the application. This step can actually begin before the development stage but is finalized in the beginning of development.

B. Coding

The bulk of Bootstrap's effort on most projects is the actual coding and development of the application. We apply industry-standard best practices including code commenting, code review by tech leads, and version control. We also develop unit tests during development that can be automatically run as part of any build to quickly verify that functions are working correctly and requirements are met.

C. Modular Releases

During the development process we schedule modular releases of function groups. This allows us to test these functions earlier and allows for DVA to review the application and provide feedback as it is being developed. Longer projects may have on the order of 4 or 5 modular releases before a final QA release.

IV. Testing & QA

Our Quality Assurance Team (QAT) continuously tests all of our code throughout the development process. Through this formalized process, we test at each milestone. Early detection of any potential problems allows us to complete the development process quickly, thereby minimizing costs. We generally include:

- Stress and load testing
- Functional (human) testing
- Automated unit tests

Of course, once the coding of a project is completed, our QAT will thoroughly test the entire application systemically. When the QAT is confident all aspects of the application will meet with the client's expectations, we will deliver the project to the client for the acceptance phase of the project. The client's feedback is a critical part of this process. This is our opportunity to perfect the application once the client has had the chance to use it.

V. Iteration

No software application is static. Even before a release is promoted to production, planning often begins for the next release and Discovery & Design begins anew. Our team of business analysts, project managers and technical leaders will work with clients to ensure that the functional and technical design of their application keeps pace in an environment where businesses and technology are constantly evolving.

VI. Software Development Life Cycle

Bootstrap employs a complex yet flexible software development lifecycle, with the client involved at each critical step of the process. During initial development, and also during future development and maintenance, code will be promoted incrementally to a QA environment which the client will have access to and can begin to "play" with the application as it takes shape. This provides the client with as much "up stream" involvement and interaction with the application as possible. The sooner issues can be discussed and

resolved, the more cost and time effective these changes become because they were caught early on in the process.

Bootstrap uses this QA environment for testing as well. During the lifecycle, once code has been fully tested and vetted by Bootstrap, the changes will be promoted to a Staging environment, where the client will have final authority to accept or deny the changes to the product prior to release to production which is yet another distinct, separate environment. The staging environment will be a close replica of the production environment to allow for smooth transitions between environments. Specifics of this process will be worked out and finalized with the client prior to development beginning. This gives the client and Bootstrap a clearly defined and vetted approach to code deployment.

In order to track changes as well as bugs in the system, Bootstrap employs its Mantis system. This is a password protected environment that both the client and Bootstrap's team will have access to. The system provides easy change tracking and allows for a transparent view from both the client's and Bootstrap's perspective at any time to see what outstanding items/issues exist. Mantis is a web application and can be accessed via username and password from any location.

VII. Maintenance

Bootstrap is committed to the ongoing support of its software. All code comes with our guarantee, and we will provide all necessary technical support to ensure our clients' satisfaction. We generally have monthly support agreements with our clients that include hosting, security services, 3rd-party software maintenance and upgrading, troubleshooting, minor application changes and availability of technical resources. Larger support contracts may include dedicated resources and weekly or daily status calls.

For more information, contact:
Michael Stineman
212.871.2020 ext. 111
mstineman@bootsoft.com