Development of Intranet

|  |  |
| --- | --- |
| Managing the design, development, implementation, and operation of a corporate intranet can be a long, difficult, and time consuming task. In this article, we present the primary steps to ensure a successful intranet development effort. | |
| Problem Definition | | |
| http://www.horsburgh.com/images/blank.gif | The definition and recording of the problem to be solved is one of the most often overlooked step of any development effort. A problem needs to be solved, so the tendency is to jump right in and solve it. For small, negligible cost efforts this is fine. For Intranet design, ignoring this step can lead to disaster. Write down and widely publish the answers to the following questions, and all other questions that are appropriate for your specific effort. Remember to keep the questions targeted to DEFINING the problem NOT solving it.  Do I Need an Intranet?  This is an obvious question, but should be taken seriously. For some businesses the answer is an easy yes, but for others, there may be better solutions. It is wise to seek professional advice when answering this question. Having an outside professional examine the question may cost some money up-front, but they are far less costly early on the development.  What specific Problems will it solve?  Write down the four, five, ten, whatever, number of problems that having an Intranet will solve. The problems should be clearly stated, be very specific, and have testable criteria for success. Make sure you publicize these problems and get user and management feedback.  What are my available resources (time, money, and personnel)?  Knowing what your actual resources are at the beginning is critical for defining the development path. If your budget is low, consider down-scaling the effort. If time is short, consider using off-the-shelf products extensively. If your personnel resources are thin, consider outsourcing. Being realistic about your actual resources will help you prevent overruns and project disappointments. Promising a gold watch when you only have resources for a plastic toy will always doom a project. Also, don't be afraid to tell upper management that the resources are too small for solving the problem. Believe me, they would rather know up front than get a surprise during deployment.  What criteria will I use to measure success?  This is an often overlooked step in the problem definition. For every problem stated, you must define a means for determining the success of the solution. If you can't think of a success criteria, then the problem is not defined specifically enough. Stay away from problem statements such as "The network must be faster." Restate the problem in quantifiable terms, like: "The network must provide a response time of no longer than 1.5 seconds for the XYZ accounting program for up to 50 simultaneous users."  Should I outsource all, some, or none of the development and operation?  If you have in-house personnel that are under-utilized or have time to be assigned to the development process, then keeping most of the development in-house makes sense. If not, then you can either hire additional staff or outsource some or most of the development. I recommend that you do not outsource all of the development. You must have some in-house expertise available or at least strong upper management support. Otherwise you may end up with a very nice system that does not solve your problems. Strategic outsourcing makes sense in most medium to large development projects. The outsourcing contractor can supply the needed expertise and personnel at the various development phases. And when a particular phase is finished, you are not left with a staff member looking for something to do. You will probably find the up-front costs of an outsourcing firm to be higher than hiring in-house personnel. But the long-term savings will be far greater with a professional outsourcing firm than by retaining in-house personnel. Remember to make sure you feel comfortable with the outsourcer's style and abilities. You will be working with them very closely. Don't just choose the largest or best-known source. How you and your outsourcer "mesh" is far more important than their list of clients.  Am I upgrading an existing system, converting from a legacy system, or developing from scratch?  Developing a system from scratch, as strange as it sounds, is by far the easiest. If you are in this situation, count your blessings. If not, upgrading an existing system or converting from one or more legacy systems will be your lot. Fortunately, you will have a long list of "things that don't work right" to begin with. Make sure that you fully understand what systems will still be in place after the migration and how they will be integrated into your intranet. If your budget is low, then consider using middleware and "web-like" products to layer on top of the existing system. With a more moderate budget, you can replace inefficient systems with newer and more powerful ones. Remember that computer hardware is cheap. It's the software and operations that are expensive. Powerful hardware can make even today's bloated software work faster. With a higher budget, consider replacing inefficient or outdated portions of the intranet with newer streamlined hardware and software. If you are not sure what the "latest and greatest" intranet products are, hire a professional intranet consultant. Their fee will be well worth it. |  |
| Requirements Analysis | |  |
| http://www.horsburgh.com/images/blank.gif | Performing a requirements analysis is critical to the success of any project. Without a clear goal in mind, success is dubious. There are a number of different philosophies about requirements analysis: top down, bottom up, inside out, etc. The method I have found to work the best is as follows:   1. Clearly state the problem(s) you wish to solve. 2. Identify the users of the completed system. 3. Formulate a specific budget -- time, money, personnel. 4. Ask identified users to specifically state what they expect the system to do. 5. Ask management to specifically state their success criteria. 6. Separate their requirements from their "desirements." Only design to requirements. The enhancement phase is where you address the "desirements." 7. Group and "bubble-up" requirements. 8. Generate a prioritized requirements table listing the requirement, where it came from, the success criteria, and priority. Keep this table high-level. A table with a dozen requirements will be much easier to manage than one with hundreds. 9. Produce a detailed development schedule including hardware, software, personnel, documentation, and reviews. Include outsourcing requirements and long lead-time items. 10. Get a sign-off of the requirements, resource allocation, and schedule from top management before you go any further.   Note that items 4 & 5 will be asked throughout the development cycle since their responses will change when they see prototypes and when they are being trained. Be sure to update items 6 through 9 each time.  Beware of getting caught in the cycle of  **You**: What are your requirements?  **Them**: I don't know, what can you do?  It's always best to ask very specific questions. Don't worry if their responses change each time you ask the question. It will happen, so plan for it. |  |
| Design & Prototyping | |  |
| http://www.horsburgh.com/images/blank.gif | There are many design methodologies. The ones I've used most successfully are 1. Rapid Prototyping (for small to medium projects) and 2. Structured Development (for large or very complex projects).  Rapid Prototyping  There are five keys to a successful rapid prototyping methodology:   1. Assemble a small very bright team of programmers, hardware technicians, designers, quality assurance technicians, documentation and graphic artist specialists, and a single manager. 2. Define and involve a small "focus group" consisting of users (both novice and experienced) and managers (both line and upper). These are the people who will provide the feedback necessary to drive the prototyping cycle. Listen to them. 3. Generate a user's manual and user interface first. You will be amazed at what you will find out by producing a user's manual first! 4. Use tools specifically designed for rapid prototyping. Stay away from C, C++, COBOL, etc. Instead use tools such as Visual Basic, HTML authoring, and similar development environments. 5. Remember a prototype is NOT the final application. Prototypes are meant to be copied into production models. Once the prototypes are successful, then begin the development processing using development tools, such as C, C++, Java, etc.   Structured Development  When a project has more than 10 people involved or when multiple companies are performing the development, a more structure development management approach is required. Note that rapid prototyping can be a subset of the structured development approach. This approach applies a more disciplined approach to the intranet development. Documentation requirements are larger, quality control is critical, and the number of reviews increases. While some parts may seem like overkill at the time, they can save a project from overruns, especially late in the development cycle. |  |
| Development & Documentation | |  |
| http://www.horsburgh.com/images/blank.gif | Once the requirements analysis is well underway, the prototypes are working, and the focus groups are becoming happy, it's time to begin the development. Coordinating hardware and software purchases and upgrades, network and hardware installation, software development, documentation guides and manuals, reviews, and testing can become a full-time job. The key to keeping a handle on all of this to maintain a good written schedule that everyone can view and to have periodic "all-hands" reviews. Remember that working with vendors can be a frustrating experience. Hardware incompatibilities, software bugs, late deliveries, mistaken cabling requirements, etc. are more the norm than the exception. Outsourcing can help, but you must be continually involved to ensure success. |  |
| Test & Review | |  |
| http://www.horsburgh.com/images/blank.gif | Testing and Reviews take place throughout the development cycle, including prototyping, development, deployment, operations, and enhancements. It never ends. It's wise to place a single individual in charge of testing and reviews. This is not a popular job, but it is critical for developing a system that works and meets each of the requirements. Be sure to empower this person (usually a quality assurance engineer) with the appropriate authority. Also, provide them with an appropriately sized staff. Testing is time consuming, tedious work and preparing for reviews and analyzing results can take much longer than you might think. Fortunately this person can save you from being surprised at budget review time and usually catches most problems before they become too big. If you outsource this task, make sure that you make it clear to the others on the team what the outsourcer's role is and what level of authority they have. |  |
| Deployment & Training | |  |
| http://www.horsburgh.com/images/blank.gif | OK, the development is complete, quality assurance is satisfied, the documentation is ready, and all the "off-the-shelf" products have arrived. Now it's time to put everything together. This can be a highly disruptive time. Make sure that you have full management support and that they understand the nature and effect of the installation and deployment disruption. Scheduling training sessions concurrently with the installation can be an effective use of time. Don't skimp on the training. Make sure you have training in the budget from the beginning and don't dip into it. The best way to ensure success is to effectively train the users so that they will actually use the system and possibly sing its praises. Also remember that training is ongoing. New employees or employees being moved or promoted will need to be trained. Each time enhancements are added, new training sessions must be scheduled. |  |
| Operation | |  |
| http://www.horsburgh.com/images/blank.gif | Intranets usually contain one or more servers. Tasks such as backups, bug fixes, software updates, hardware maintenance and upgrades, print and media services, electronic mail account maintenance, security patches, and other similar tasks must be performed regularly. Operation and maintenance of such services require an operations staff. It is not enough to "let the users take care of it." If you are providing these services in-house then you will need on-site support from either an outsourcing agency or in-house staff. The current trend is to outsource most of these services including the actual servers to an intranet outsourcing firm. Outsourcing can result in a substantial savings. Just make sure that your provider can supply the services you require and is available when you need them. Also, be sure to discuss security requirements with them before you hire them. |  |
| Enhancement | |  |
| http://www.horsburgh.com/images/blank.gif | There is always one thing you can count on: "Requirements Creep." The more successful the system, the faster requirements creep will occur. As your users become more sophisticated they will want more and more capabilities. If you can respond quickly and efficiently, your users will again sing your praises (and upper management will definitely take notice :-). Make sure that you have designed in the ability to add features from the very beginning. Remember to design-in scalability and flexibility at all phases of development. |  |

|  |  |
| --- | --- |
|  |  |

**Intranet/Extranet Benefits**  
Intranets and Extranets offer corporations, small businesses and organizations many benefits:

* **Increased productivity** – Intranets and Extranets are opening new opportunities for corporations and businesses to communicate and operate more efficiently. Employees can be more efficient with up-to-the-minute information at their fingertips. Customers can be allowed to perform actions, such as referencing information and performing transactional business-to-business activities, thereby boosting customer loyalty, improving customer service and responsiveness, and saving the business time and money.
* **Reduced margin of error** – An Intranet and Extranet can reduce the margin of error when specific groups directly access internal applications. This could be as simple as giving employees access to human resource forms or providing customers with access to their order histories, or something as complex as processing orders.
* **Flexibility**- When information and applications are directly available to employees, partners and customers, everyone can work when and where it’s most convenient. This self-serve approach improves efficiency, and eliminates the costs associated with in-person information exchanges. For example, a company can provide customer service information after regular business hours through an Extranet.
* **Timely and accurate information** – An Intranet and Extranet can be used to provide accurate and timely information. They can be utilized to instantly change, edit and update information such as human resource forms, price lists or inventory information.
* **Faster time to market** – An Intranet and Extranet can help a company to get products faster to market by using it as an internal project management system. By making proposals and specifications available to suppliers, and by giving customers and partners up-to-date information on current projects, development time can be reduced significantly.
* **Reduced inventory** – One of the hallmarks of a business-to-business Extranet is its impact on supply-chain management. By connecting an inventory application directly to a supplier system, companies can process orders as soon as their system knows when they are needed, thus reducing inventory and making the procurement process more efficient.
* **Greater customer loyalty** – Extranets enable companies to establish and maintain powerful, long lasting relationships with their customers and business partners by making it easier to perform business-to-business transactions and to get timely and accurate information to them, whenever they need it.

**Web syndication and e-commerce**

Similar to syndication of proprietary content, web syndication has been used to distribute product content (feature descriptions, images, specifications etc.). Given that manufacturers are regarded as authorities and that most sales do not happen on manufacturer Web sites, best-in-class manufacturers take their best content and enable other retailers or dealers to publish the information on their sites. By syndicating content, a manufacturer is more likely to pass consistent and often comprehensive information to channel partners. Such web syndication has been shown to generate an increase in sales.

Web syndication has been increasingly used as a way to syndicate online news content to websites, too as part of [Search Engine Optimisation](http://en.wikipedia.org/wiki/Search_Engine_Optimisation) techniques. Adding news content lets websites target specific keywords that could cost a fortune in PPC campaigns absolutely free. Also, by the very nature of writing about a topic - for example the UK housing market - you naturally cover other keyword variations in the same page, eg. house prices, mortgages, first-time buyers - all in one page: all keywords that someone interested in the UK housing market could type into a search engine also. Done on a daily basis, over time, news content will improve your search engine ranking considerably.