Applying a User-Centered Design Approach to Data Management:

Paper and Computer Testing

Jeffrey R. Wilson, Ph.D.

Pearson

March 2008

Paper presented at the annual conference of the American Educational Research Association, New York, New York (March 2008).

Abstract

This paper discusses the application of a user-centered design (UCD) approach to a web-based application system that supports data management components of the high-stakes assessment lifecycle. In order to fully explore how a UCD approach enhances the process for developing a web-based application system that supports data management for paper and computer testing, this paper starts with an explanation of the typical data management development process before a UCD approach is applied to it. The paper then discusses how the user experience can be enhanced when a UCD approach is applied to developing data management systems. Next, the paper presents a case study for applying a UCD approach on a web-based application system that supports the paper and computer testing process. Finally, the paper discusses insights learned from applying a UCD approach to data management tools in an educational assessment setting.

Applying a User-Centered Design Approach to Data Management:

Paper and Computer Testing

There is considerable work that goes on behind the scenes to make sure that state departments of education have the necessary materials and tools to ensure that their *high-stakes* assessment process is a success. There are a myriad of tasks that educators focus on at the state, district and school level to make the testing experience successful. Since students may take state tests on paper or on the computer, both must be considered when helping states administer *high-stakes* tests. For example, when testing students on paper, districts and schools must order paper materials, ensure that the materials will arrive on time, place additional orders when their initial order doesn't meet their needs, upload student data so that students' demographic data are correctly printed on labels for their tests, and have access to student test results. When testing students on the computer, districts and schools must make sure that students are assigned to the correct test, the students must be put into test sessions so that they can take their tests on the computer at their scheduled time, and that there are enough computers for each test session.

This paper discusses the application of a user-centered design (UCD) approach to developing a web-based application system that supports data management components of the high-stakes assessment lifecycle for both paper and computer testing. User-centered design has been described in many ways. Norman (1998) provides a comprehensive description.

> It's a process of product development that starts with users and their needs rather than with technology. The goal is a technology that serves the user, where the technology fits the task and the complexity is that of the task, not of the tool. At its core, human-centered product development requires developers who understand people and the tasks they wish to achieve. It means starting by observing and working with users, writing a simple instruction manual—no more than one page long, if possible. It means constructing and evaluating physical and software mock-ups of the device to see how its potential users employ it for its intended activities, allowing all to judge whether the design meets their requirements. Then, after a period of iterative evaluation and redesign, it means building a technology that fits the mock-ups.

In order to fully explore how a UCD approach changes the process for developing a system that supports data management for high-stakes paper and computer assessments, this paper will start with an explanation of the typical data-management development process. The paper will then discuss how the user experience can be enhanced with the application of a UCD approach to developing data management tools. Next, the paper will present a case study for applying a UCD approach on a web-based application that supports both the paper and computer assessment process. Finally, the paper will discuss insights learned from applying a UCD approach to data management tools in an educational assessment setting.

Assessment-related Data Management

First, consider a hypothetical scenario where there is a need to develop a web-based application to support *high stakes* assessment. Pearson wins a contract for a particular state to support both paper and computer testing for the fall of 2008. If Pearson already has contracts with that state then there is a specific team of dedicated professionals (i.e., a program team) within Pearson who are already focused on supporting that state. The program team has developed relationships with key stakeholders at the state's department of education (DOE) to better understand what they need to deliver for districts and schools for each assessment. As the program team gathers requirements about how they need to support that state for each assessment, they pass the information along to a technology organization within the Assessment and Information group at Pearson. The technology organization then develops or modifies an existing web application to support the program team, the state, and the people at the district and school levels. Once the web application is ready for review, the program team allows the stakeholders at the DOE to review the application. If it meets their needs, then the system is

made available for use. If the system needs to be enhanced, then it is re-engineered for the DOE. Or the system is released before all the desired changes are made and it is modified as time and schedule permit.

Of course, the process is rarely that simplistic. For example, if one now imagines that same process for 20 different states, it becomes apparent just how busy the technology organization becomes to deliver systems for its various customers. In other words, deadlines are critical, delivery of a system is crucial, and delivering a system that meets the needs and expectations of its different users is imperative for a positive user experience. Given all of this, a fair question to ask is, "Can we build a system that basically works for all of our customers (minus customer-specific customizations) in an efficient manner that will provide them with a satisfying user experience while reducing the amount of re-work once the system is released for general consumption?" If the answer is *yes* or even *maybe*, then the follow-up question is, "What do we need to do to build *that* system?"

Applying a UCD Approach to Assessment-related Data Management

Creating a compelling and satisfying user-experience isn't only the concern of usability professionals who focus on UCD. UCD is a process that all members of a development team engage in to ensure that a system's users have a positive experience. A corollary to the above point is that there are many groups within an organization (e.g., Help Desk) that are focused on the *user* and customer satisfaction. Therefore, it is misleading to assume that only usability professionals care about the user experience of a product (even if a UCD process isn't employed in an organization).

However, usability professionals provide expertise by helping an organization deliver an optimal user experience by seeking to use UCD processes and methodologies in product development. Even if product development isn't focused solely on UCD as their primary development process, UCD can complement the product development lifecycle by providing knowledge about the people using the system (e.g., personas) and designs or heuristics that consider the needs of those users. Therefore, consider these questions again:

- Can we build a system that basically works for all of our customers (minus customer-specific customizations) in an efficient manner that will provide them with a satisfying user experience while reducing the amount of re-work once the system is released for general consumption?
- What do we need to do to build *that* system?

A UCD approach need not necessarily be applied to get a *yes* answer to the first question although more than likely the development approach includes different elements of UCD. However, for the second question, the central argument of this paper is that it *would be prudent and beneficial* to apply a UCD approach when building a product in order: a) to provide a framework for design decisions and; b) to include some specific methodologies and evaluations of design decisions that focus on the end-user.

What is an example of a UCD process? Rubin (1994) provides a simple example of a UCD-focused product-development lifecycle: a) User and usage needs analysis; b) Specification of requirements; c) Preliminary design; d) Detailed design; e) Product build; and e) Product release. Therefore, to apply a UCD approach when developing a web-based data-management system to support high-stakes assessment one must determine: a) several research questions that would identify the target users and their needs; b) the requirements for the system; c) the preliminary

designs that would need to be created (with high-level architecture and navigation considerations along with common conventions in web design); d) the number of iterations on designs that would need to be pursued in order to create detailed designs for a solution that would be effective and efficient as well as technically feasible. Once detailed designs are finished product development could focus on building the product for release. Thus, we can determine two major areas for consideration before a product is built and released for people to use: 1) User research to understand the target users, their needs, and the requirements of the application and; 2) design that includes both: a) preliminary design—to determine if the design is meeting the needs of the users and addressing their requirements for the application and; b) iteration of design through evaluations to increase the likelihood that the application meets the needs of users, matches their expectations, and can be effectively and efficiently used by end-users once the system is released. Let us examine each of these areas in more detail.

User Research. Before building a web-based application that will be effectively and efficiently used by people, there are several research questions that need to be answered:

- Who are the target users for different work tasks?
- How do those people complete their work tasks?
- When do they engage in their work tasks?
- How frequently are they doing these tasks?
- What are the products they currently use to complete those tasks?
- What are the good points and pain points with current products and processes?

It is critical to understand who will be using the application and how those users expect to use the application given their job functions and work tasks. It is also important to understand the environment and conditions in which users work because that determines the way a particular

application is designed for them to use. Finally, by examining the current applications people use to complete their assessment-related tasks, a wealth of information can be obtained about the experience with those applications. Thus, the positive experience that people have with those applications should be replicated or expanded upon in a new application and the pain points that people have with those applications should be reduced or eliminated in a new application.

Design. After the research questions are addressed, an initial design framework for the new application can be developed. This design framework considers the preliminary user research, examines common conventions in products, services, and web sites/applications, and keeps in mind the considerable research and design heuristics that exist for designing products and services. For example, Galitz (1997) provides a comprehensive list of industry-standard design goals for a product.

- Aesthetically Pleasing
- Clarity
- Compatibility
- Comprehensibility
- Configurability
- Consistency
- Control
- Directness
- Efficiency
- Familiarity
- Flexibility
- Forgiveness
- Predictability
- Recovery
- Responsiveness
- Simplicity
- Transparency
- Trade-offs

Although a detailed definition of each of the heuristics on Galitz's list is beyond the scope of this paper, the list shows that there are numerous factors to consider when building a usable product besides conducting preliminary user research and considering common design conventions.

Once the design framework is in place then preliminary prototypes can be created and evaluated against the design framework. Therefore, issues like the architecture of the system, navigation of the system, page types, page layout, and page elements are all considered during the initial design phase.

However, UCD considers iteration a key to building successful applications. Even with careful planning, up-front user research, and consideration of the extant research and common design heuristics, the design of an application is untested unless there is some systematic method for evaluating early prototypes and then using that data to continuously improve upon the design of the application. Therefore, the goal is to iterate on designs so they will be more effectively and efficiently used by people.

In the early stages of design, considerable work is done evaluating prototypes internally and making sure that the preliminary designs are technically feasible and follow the design framework. However, it is also important to test the designs with actual users by having them complete common tasks or review the prototypes to ensure that the designs meet their expectations. Again, whether it is internal evaluations or collecting data from actual end-users, the goal is to iterate on designs to improve the likelihood that the application will satisfactorily meet the needs of the end-user. Usability studies takes advantage of preliminary user research to determine who the target users are for the system and their major work tasks. The objective is then to observe target users completing their key work tasks and to identify the problems that would keep them from completing those tasks successfully and quickly with high satisfaction. Thus, a UCD approach requires ensuring technical feasibility while meeting user expectations through evaluating the design framework.

Assessment-related Data Management: A case study

Creating PEMSolutions is a case study that illustrates an attempt to follow the UCD framework for the development of a new product. Historically, the Assessment and Information group at Pearson had developed two separate products to meet the needs of their customers: SchoolHouse and eMeasurement Services. SchoolHouse (see Figure 1) was created to assist users with high-stakes paper testing and eMeasurement Services (see Figure 2) was created to assist with students taking tests on the computer (online testing). Both products considered their users and modified the user experience as feedback flowed in from people using the products. Therefore, the products were focused on customer experience but that doesn't necessarily mean a UCD approach was applied to the development and evolution of those two products.

Insert Figures 1 and 2 Here

To maximize the user experience and decrease software maintenance costs, it was decided to take these two existing products with different user experiences and combine them into one product—PEMSolutions—(see Figure 3) that would deliver paper and online testing services—with the long-term goal of being able to continue to integrate other existing functionality into this system as well. The first state to utilize PEMSolutions was a technologically advanced state who knew that PEMSolutions was being built to serve all existing and future customers, yet would provide them with the integrated solution they needed for their large-scale assessments.

Insert Figure 3 Here

User Research. It was beneficial having existing products with a history of supporting high-stakes assessment (compared to starting from scratch) since the mission-critical tasks for supporting high-stakes assessment were already known. Therefore, in the beginning, we asked the following questions—*Who are the primary user populations completing high-stakes*

assessment-related tasks and *What are the primary users' work practices and workflow, their concerns, and their experiences with their current high-stakes assessment-related products?*

A survey was conducted across 10 states (since we wanted to create a system that would work for different states and not just build a custom solution for one state) to find out who was responsible for the different assessment tasks that occur for an end-to-end testing scenario (both paper and computer tests). Although many different user groups are involved in high-stakes assessment (e.g., in smaller districts people have multiple roles), the results from the 127 survey respondents showed that district test coordinators are the primary user group that completes most of the high-stakes assessment-related tasks. For example, district test coordinators typically complete the following assessment-related tasks:

- Entering enrollment counts for ordering paper test materials
- Ordering additional paper test materials
- Uploading the file of student information (Pre-ID) for paper tests and electronic tests
- Creating electronic test sessions
- Making corrections to student information after it is uploaded (for paper and electronic tests)
- Signing for delivered paper test materials
- Unpacking and distributing paper test materials
- Gathering and returning paper test materials

There are other user groups that complete the same tasks as district test coordinators (e.g., school test coordinators create electronic test sessions too) and different tasks (e.g., teachers are the primary proctors), but district test coordinators performed most of the tasks that required planning, coordination, and training. Interestingly, district test coordinators also tend to be the gatekeepers and experts for the school-level personnel that use the system and they are the

primary communicators back up the chain to the DOE and to Pearson about the user experience. Therefore, it was important to gather more information about district test coordinators, their current work practices and workflow, and their feedback about their user experience with existing products in order to understand how to design a new system.

After the initial roles and responsibilities survey, follow-up interviews were conducted with 21 district test coordinators from several different states to understand their work practices and workflow, their concerns, and their experiences with their current high-stakes assessment-related products. Three main themes emerged from the interviews:

- **No time**. District test coordinators are incredibly busy and strapped for time. This was important information to find out when considering design options. District test coordinators don't have much time for inefficiency or time *to figure things out*. Because they are engaged in making sure that high-stakes assessments are accomplished without a problem, district test coordinators need to be able to complete tasks successfully and quickly and need information that increases their confidence that their various work tasks are progressing as they expect them to progress to completion.
- **More control**. Whether it is ordering test materials, receiving test materials, sending student data files, or other assessment-related tasks, district test coordinators want to be in control of their work tasks and know how their various work tasks are progressing toward completion.
- **Infrequent use**. District test coordinators engage in many work tasks and only use their assessment systems when they need to perform an assessment-related task. For example, a district test coordinator may only order test items twice a year. Also, a district test coordinator may be using more than one vendor depending upon the assessment. Therefore, taking advantage of designs metaphors that are common and providing aid in a variety of ways in the system to help people complete tasks and feel like they are in a familiar environment is important.

Design. After completing the initial research, brainstorming an initial design was the next step in the process. There are a variety of ways to create an initial design. Because we had two existing products, one of the first things we did was capture all the functionality that was provided in both products. We then examined where that functionality lived in the existing products. For example, was the functionality part of a mission-critical task that was readily available compared to a secondary support task that was housed a couple of levels down in a system? We also started to mock-up initial concepts that displayed an information architecture based upon current products, our initial research, common web conventions and design heuristics, and a respect for the possible evolution of the product given some of the long-term product strategy goals that were stated by senior management.

To be clear, creating a new design for a product that allows users to complete tasks effectively, efficiently, and produces positive customer satisfaction is an incredibly difficult effort. Even with all the above-mentioned tools at our disposal, there were a multitude of design suggestions to achieve those goals both at a high-level and down to specific behavior for how a function should be designed for our users. Also, there were considerations with regard to technical feasibility of a design. However, without the initial data and planning, this process is even more problematic.

Further, even though design heuristics are valuable, they often compete with one another. For example, a common design heuristic is that users should feel in control of the system and know where they are and what they are doing in the system. Also, users should be able to recover from potentially catastrophic errors. However, efficiency is important because users also like to complete tasks as quickly as possible. A simple example of a dilemma here is the use of confirmation screens after users complete tasks. For example, if a user wants to delete a student

from a system, which is a serious task, it would be easy enough to just delete the student once the user clicks on a delete button. However, what if the user clicked this button by accident? What if the user wanted to know whether the student was actually deleted from the system? By simply asking users if they *are sure* that that they want to delete a student and then following this message (assuming an affirmative response) with a confirmation screen that the student was deleted, users now have complete control of the process and knowledge that their task was completed successfully. However, this process would entail two extra clicks: 1) to answer the "Are you sure?" message; and 2) to dismiss the confirmation message. This example is typically referred to as a *design trade-off*. Therefore, when making initial design decisions, design trade-offs are typically applied with regard to emphasizing different design heuristics, applying extant research, focusing on the requirements of the system, and considering the research learned about key user populations.

Iterative design involves both internal design reviews and collecting data by observing people work with and review the application. Usability studies provide a wealth of information showing whether a proposed design meets the needs of users to effectively and efficiently use the system to complete tasks. Before conducting usability studies with target populations, key internal stakeholders (e.g., program team members) were asked to participate in usability studies. Program team members know their target populations and also know how their stakeholders at the DOE might react to proposed user experience solutions. Also, program team members and other stakeholders are often experts with considerable experience and indispensable domain knowledge. Therefore, they provide a wealth of information about whether our design will likely meet the client's expectations. Also, having the same domain knowledge as the customer, if they

can't complete tasks effectively and efficiently then it is likely that the key target users will also struggle to complete the tasks effectively and efficiently.

After collecting data from internal program team members, it was then imperative to collect data from target users of the system. Two different studies were conducted with target users on key tasks with the new system. This data was then evaluated and used to modify aspects of the system to increase the likelihood that the target users would not encounter the same types of problems noted during the usability studies. For example, in one study, ten district test coordinators from five different states were asked to complete the following tasks in a prototype:

- Upload/Verify Data Files
- Filter Student Records
- Add/Remove/Edit Student Data
- Create Student Groups
- Order Test Materials
- Create/Modify Test Sessions (for online tests)
- Check Order Shipping Status

Of course, the major emphasis in a usability study is to observe target users completing representative tasks. Therefore, after watching users complete the above tasks with the prototype, usability issues were identified related to completing these tasks—some specific to the task and some more general observations about the system. However, the major usability issues that were observed fell within the following categories:

- **Navigation and Discoverability**. For example, users tended not to select a test administration and often did not notice if the test administration was incorrect. Also, users tended not to notice the sub-navigation links under the main navigation tabs.

- **Page Behavior and Forms**. For example, users did not understand that "Save" or "Next" must be clicked to submit changes entered into forms. The relationship between filtering dropdowns and the filter buttons was not clear to some users.
- **Status and Confirmation**. For example, users did not immediately understand what sequence of activities was required to complete a workflow in the prototype. Users wanted more status and confirmation information to verify that tasks were completed successfully.

Based upon the analysis of the usability problems identified, recommendations were made to increase the likelihood that users would be more efficient and effective using the system, the prototype was updated in various ways, and a follow-up usability study was conducted with users. For example, between the first and second usability study the usability of the navigation was examined. During the first usability study, it was clear that users didn't readily see the sub-navigation under the different tabs which reduced the likelihood of those users completing tasks effectively and/or efficiently. Since these users have to complete critical tasks successfully and quickly, the tab headings and major tasks under each tab were replicated in the content area of the Home page. After making this change, users were observed using these new links successfully and quickly on the Home page to navigate to their mission-critical tasks.

Insert Figures 4 and 5 Here

Within Pearson customer reviews are typically conducted with a production system. However, when applying a UCD approach to customer reviews, the goal is to let key personnel at the DOE level review prototypes to see if the prototypes meet their expectations. Using prototypes helps to clarify requirements for the system, pushes the review process up to an earlier date which gives time for a corrective course of action, and sets the expectation that the

prototype isn't necessarily the final product. Also, these reviews afford an opportunity to explain design rationale and to emphasize that usability studies were conducted with key target populations to ensure the design of the system is usable. During the development of PEMSolutions, several customer reviews were conducted with members of the DOE using prototypes. This kind of review received positive support and allowed on several occasions for the DOE to provide feedback that made it into the system before the system was released for use.

The UCD process doesn't end once a system is released for production. Once a system is released for customer use, the UCD process begins again. However, once a system is released it is important get feedback from people that use the system and then use that feedback to make changes to a system. There were a variety of methods to obtain user feedback after the release of PEMSolutions. First, the program team received feedback from the DOE and from people at the district and school level. Second, users called our Help Desk to get answers to questions about using the system. Third, customer interviews were performed to better understand the issues that existed using the system. Finally, after using the system during an assessment period, satisfaction surveys were employed to better understand what parts of the system were easy to use and what parts of the system needed to be examined for further improvement.

Satisfaction surveys are critical for getting more than just anecdotal data on the user experience of a system. Also, satisfaction data can be used for creating a post-release evaluation framework for better understanding anecdotal data that comes from the users of the system. For example, a PEMSolutions satisfaction survey was sent to 137 people. Participants were asked to answer the survey and/or forward the survey to other relevant people in their division. Fifty-four people responded to the survey. Overall, most respondents reported a neutral to positive experience completing tasks using PEMSolutions. While there were several areas of the system

that were considered easy to use; there were other areas of the system that didn't receive as strong support. Whether the data was examined from respondents who were neutral to positive or simply positive about completing tasks within the different areas of PEMSolutions, it was clear that there were several areas that needed to be examined for enhancement moving forward.

Feedback from users comes from a variety of sources to the program team that supports a state. However, one of the issues about getting informal feedback is that it is difficult, at times, to determine how critical an issue is for people compared to other issues reported by people. Also, what might be an issue for one person may not be an issue for other people using the system. Therefore, the satisfaction survey was used to help put anecdotal feedback into a larger context with regard to the overall user experience. Second, for areas that didn't receive high customer satisfaction ratings, we were also able to access the data the program team received from the customer to create some hypotheses about those areas in the system. Finally, the results of the satisfaction survey as well as other feedback received from the users of the system were used as a way to prioritize areas to focus on for enhancements that would provide the largest impact on enhancing the user experience of the system. Of course, as the product continues to evolve, we will continue to measure user satisfaction as part of the UCD process.

Lessons learned: Looking to the future

There is the ideal UCD approach that many authors have outlined in their books and then there is the practical reality of applying a UCD approach during a product-development lifecycle given the history of an organization where UCD is new (or even established). In this final section of the paper, some of the caveats and lessons learned from applying a UCD approach for developing a web-based application for high-stakes assessment are examined.

One of the most valuable ways to collect data about how people work to complete their critical tasks is to observe them in their natural work environment. However, selling the idea of shadowing district test coordinators as they are doing their work is not that easy because it is both costly with regard to travel expenses and scary with regard to having people be observed while they are working on such mission-critical tasks. However, we still need to collect data when trying to understand how users work or trying to understand the issues that users have working in a system.

Therefore, one of the caveats to be mentioned with regard to this case study in UCD is that instead of observing people work, we conducted interviews with people where we had them focus on describing work behaviors performed during an assessment process. On a related point, if we collected data from a survey, the program team, and/or the help desk about a particular issue with the system, instead of observing people experience this issue in real time, we would schedule time with users to better understand their concerns.

We were able to accomplish this by using a remote testing tool that allowed the users to access our network so we could see where the issues arose rather than rely on the verbal feedback alone. Thus, we have learned that applying a UCD approach in our domain has some practical limitations with regard to observing people work. However, we have also learned that applying a systematic approach to understanding our target users, identifying usability issues through a variety of means, measuring satisfaction through surveys, and using prototypes as a tool to communicate designs and requirements to both internal and external stakeholders has still provided valuable data and has likely improved the user experience.

Also, our success of applying a UCD approach to product development is related to the perceived success of the work we do within an organization. Therefore, to some degree, people

engaged in a UCD process must evangelize the process to their peers in product development to get their buy-in on different UCD activities. However, evangelizing isn't just about *talking up* UCD and its benefits related to creating a satisfactory user experience. It is also showing colleagues the value of the UCD approach by giving those people data they can use along with design rationale. Often design rationale is based upon research and common design conventions that are considered optimal for a great user experience by experts in the field. Some design rationale can be attributed to the fact that conventions are used consistently in some of the top sites or web applications. Finally, we have also learned that listening is not only essential for our customers. It is critical to listen to our developers on issues of technical feasibility and to reflect their expertise on those issues in our designs.

Of course, UCD is about iteration and not just iteration of a design. We know that improving upon applying the UCD process at Pearson is an evolution where we will continue to iterate and learn about what works and what doesn't work in our environment. As we continue to learn and iterate on our process, we hope to continue to apply UCD more effectively and efficiently in the future.

References

Galitz, Wilbert O. (1997). The essential guide to user interface design: an introduction to GUI design principles and techniques. New York, NY: John Wiley & Sons, Inc.

Norman, Donald A. (1998). The invisible computer: why good products can fail, the personal computer is so complex, and information appliances are the solution. Cambridge, MA: The MIT Press.

Rubin, Jeffrey (1994). Handbook of usability testing: how to plan, design, and conduct effective tests. New York, NY: John Wiley & Sons, Inc.

Figure 1. SchoolHouse (paper testing)

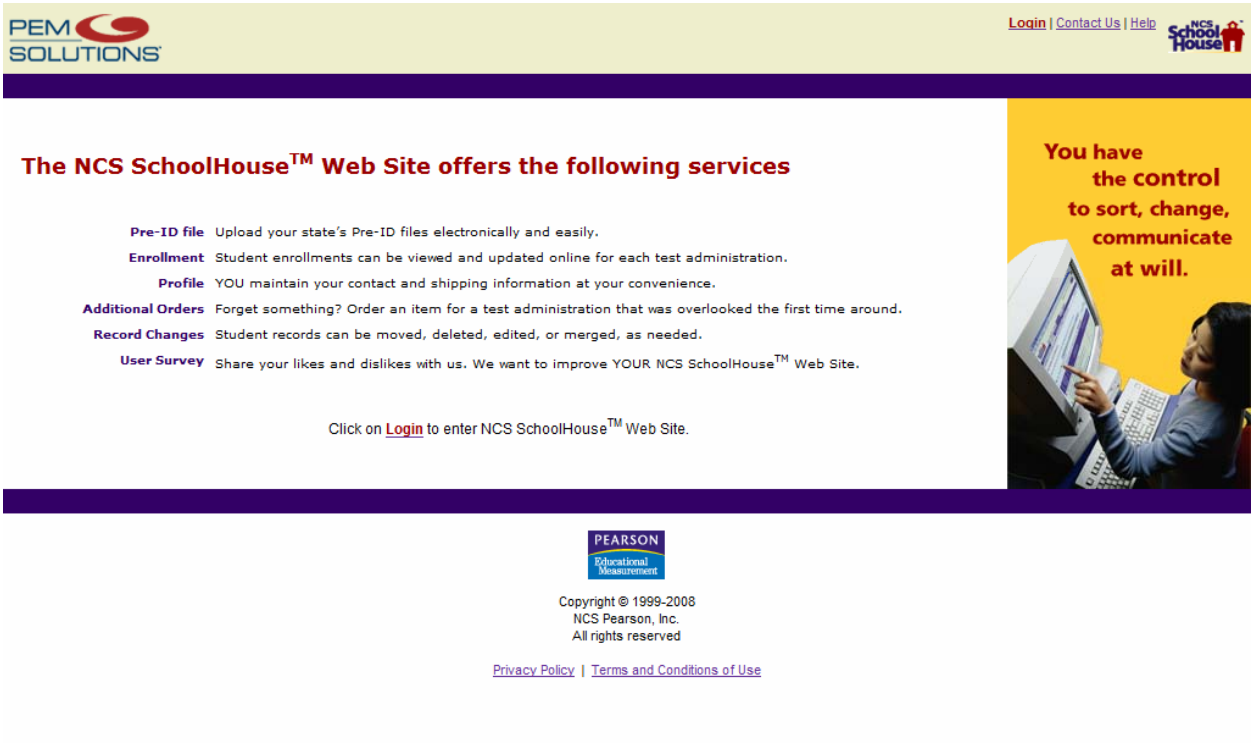

Figure 2. eMeasurement Services (computer testing)

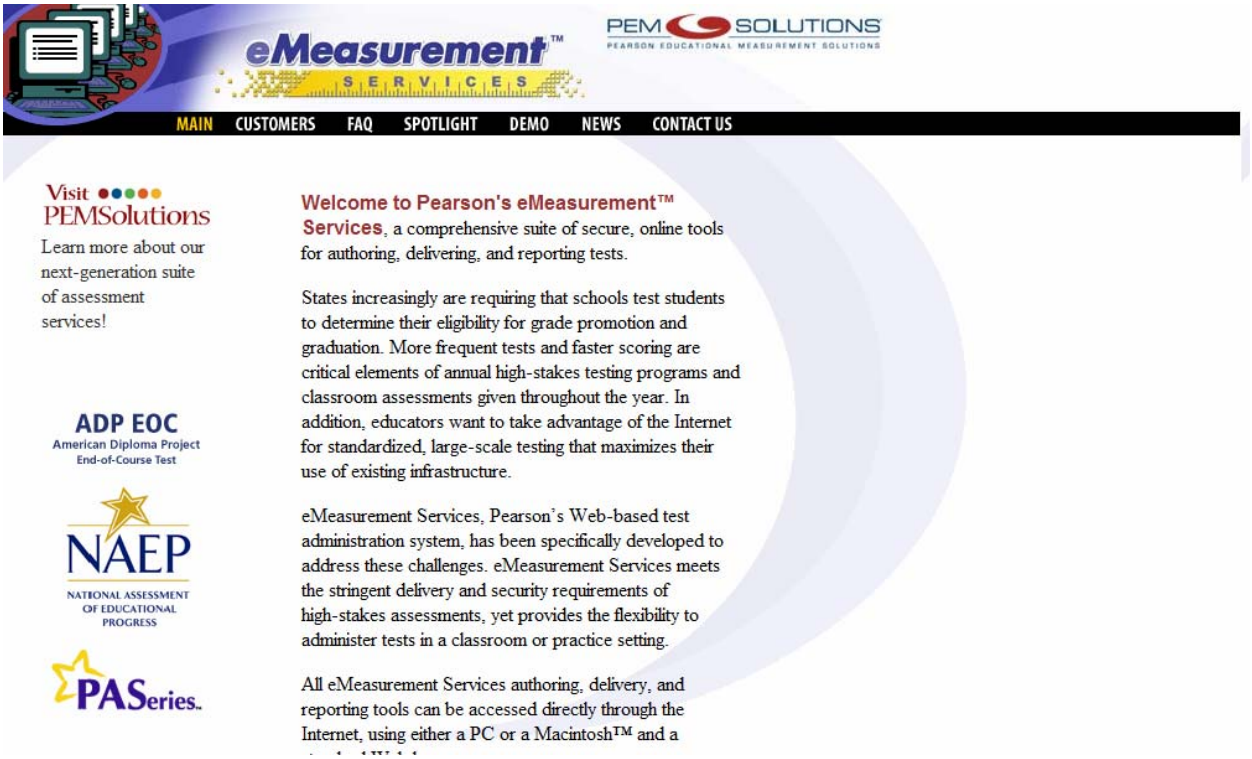

Figure 3. PEMSolutions (paper and computer testing)

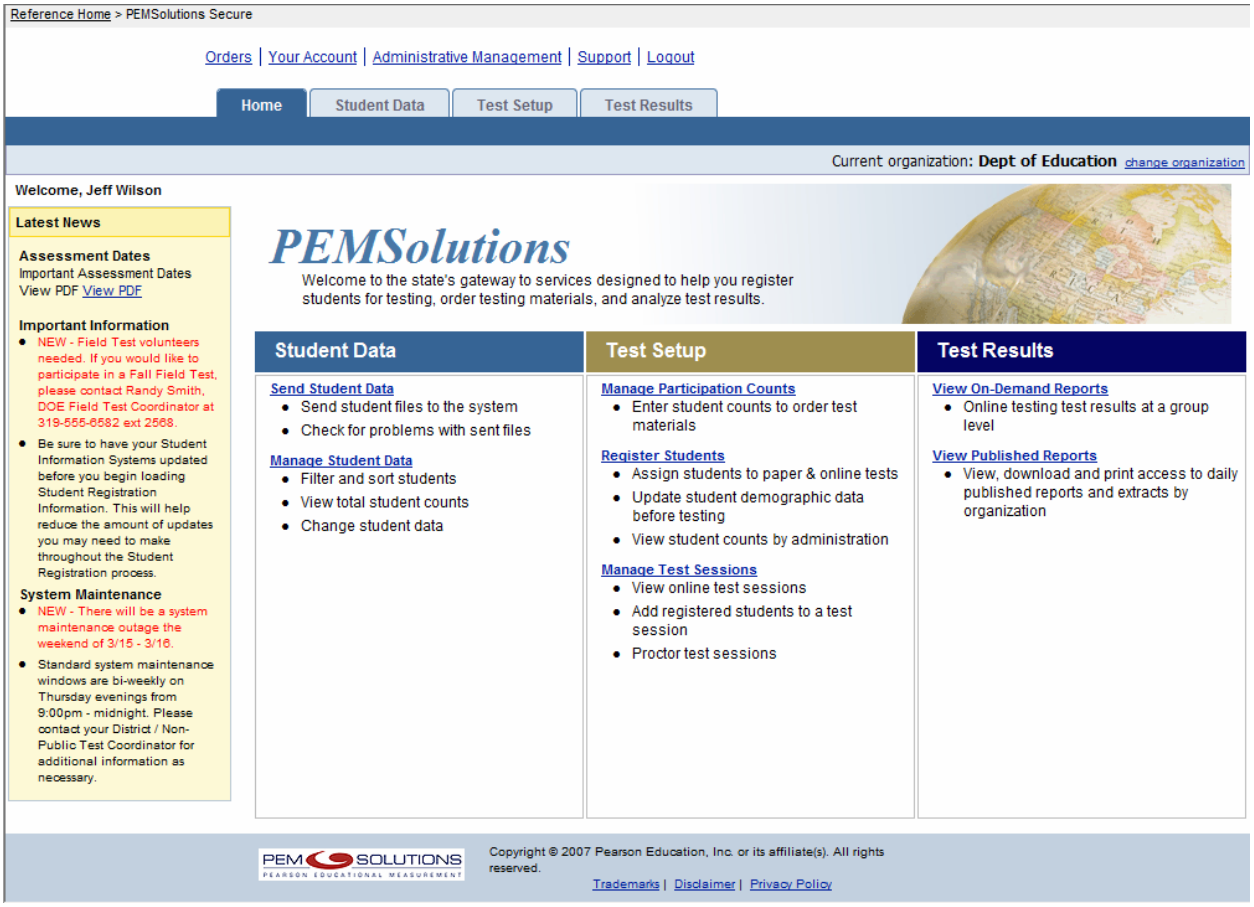

Figure 4. Early Prototype of PEMSolutions

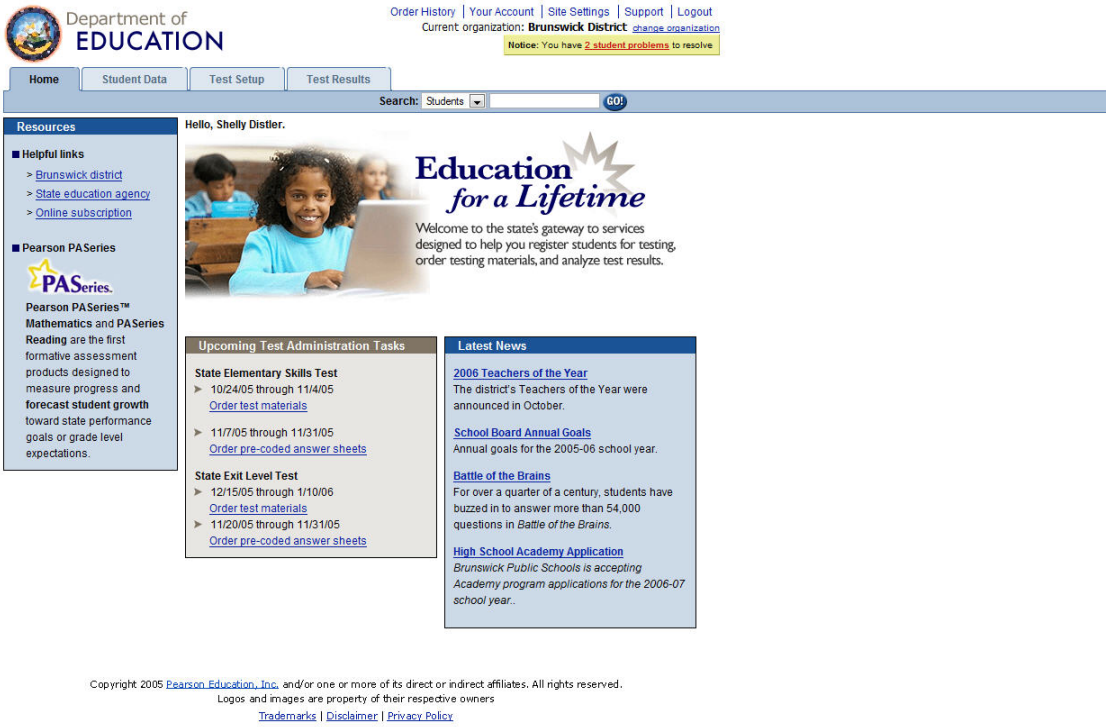

Figure 5. Early prototype of PEMSolutions after user testing