

# Wireless Data Collection of Self-administered Surveys using Tablet Computers

Kyle W. Singleton<sup>1</sup>, Mars Lan<sup>2</sup>, Corey Arnold, PhD<sup>1</sup>, Mani Vahidi<sup>3</sup>, Lisa Arangua<sup>3</sup>,  
Lillian Gelberg, MD, MSPH<sup>3,4</sup>, Alex A.T. Bui, PhD<sup>1</sup>

<sup>1</sup>Medical Imaging Informatics Group; <sup>2</sup>Computer Science;  
<sup>3</sup>Family Medicine; <sup>4</sup>Public Health  
University of California, Los Angeles, CA

## Abstract

*The accurate and expeditious collection of survey data by coordinators in the field is critical in the support of research studies. Early methods that used paper documentation have slowly evolved into electronic capture systems. Indeed, tools such as REDCap and others illustrate this transition. However, many current systems are tailored web-browsers running on desktop/laptop computers, requiring keyboard and mouse input. We present a system that utilizes a touch screen interface running on a tablet PC with consideration for portability, limited screen space, wireless connectivity, and potentially inexperienced and low literacy users. The system was developed using C#, ASP.net, and SQL Server by multiple programmers over the course of a year. The system was developed in coordination with UCLA Family Medicine and is currently deployed for the collection of data in a group of Los Angeles area clinics of community health centers for a study on drug addiction and intervention.*

## Introduction

Clinical and translational research studies necessitate accurate data collection methods to gather important information from subjects using surveys. Paper-based collection methods have been the standard approach for gathering data directly from the patient despite increased access to computer technology. Arguably, paper has proven inefficient and solutions targeted at increasing survey efficiency, such as document scanning, come at a high expense. Research in the 1980s on computer-aided personal interviewing (CAPI) laid groundwork for integrating computer systems into the survey process. Computer and auditory technologies have been used to improve telephone surveying for some time<sup>1</sup>. More recently, web-based technologies have made it simpler for surveys to be designed and given to larger groups. However, these systems see limited use for medical data collection outside of clinical trials.

The majority of current browser-based surveying systems are designed for use on desktop/laptop computers. While some populations have ready access to such technology, there are many potential situations where subjects may not have direct access to their own computer; and access to a web browser is difficult when interviewing subjects on the street or in a clinic environment, for example. Community-based studies may involve homeless and indigent populations who do not own computers or have readily available Internet connections. Some potential solutions are to provide kiosks in healthcare facilities, or to provide temporary use of a laptop. Kiosks lack portability and typically need a centralized location that reduces privacy and comfort for long surveys. Laptops, while more portable, lack touch screen capabilities that make navigation intuitive for infrequent computer users. The recent increase in touch screen computers is opening the door to improve current browser-based survey programs. Touch screen netbook computers and the emerging market of tablet computers provide a more readily understood method of data input, and hardware costs can be lower than comparable options.

In this paper, we explore a new framework for designing flexible surveys for data collection on tablet computer systems. The goal of this work is to develop a configurable open-source surveying system to aid in data capture within the outpatient clinical setting, including community-based clinics and other physical areas where electronic infrastructure may not be available. In addition, developing a survey system geared towards patient self-administration enables new data collection methods beyond the capture of data directly from case report forms (CRFs). The developed system, EMMA, incorporates a simple user interface utilizing touch screen technologies and text-to-speech (TTS) to enable less educated users to self-administer surveys. The development of the EMMA framework was performed in coordination with UCLA Family Medicine and has been deployed for their National Institute on Drug Abuse (NIDA) Quit Using Drugs Intervention Trial (QUIT) Study, a multi-clinic randomized-control trial capturing data from low-income subjects concerning their drug use.

## Background

To date, the majority of modern electronic data capture systems have focused on handling clinical trial data. A few open source examples include the Research Electronic Data Capture (REDCap) project from Vanderbilt University<sup>2</sup>; TrialDB out of Yale University<sup>4</sup>; and OpenClinica, developed commercially<sup>5</sup>. These platforms rely on traditional web architecture, where the users enter information through server-side generated HTML forms. This approach shares many benefits of standard websites, such as cross-platform compatibility, fast development and distribution cycle, and user familiarity with web browsers. These systems are specifically designed to support data capture in translational research projects from data sources like CRFs. Data input is performed by a highly trained user base of research volunteers, clinical trial coordinators, and physicians. While these systems maintain robust tools for describing data and case reports in clinical trials, most do not provide a mechanism to design surveys allowing patients to self-report data.

Modern web-based technologies for survey building and collection are designed to emulate paper survey forms in an online environment through a web browser. SurveyMonkey and SurveyGizmo are two popular websites in use by individuals and large corporations<sup>6,7</sup>, providing paged surveys with a wide variety of questions types. Using these services for long surveys and the collection of large numbers of subjects requires costly subscriptions. In addition, only SurveyGizmo currently provides a system with HIPAA (Health Insurance Portability Assurance Act) compliance. Lime Survey, an open source project, provides a similar set of tools for institutions as an alternative to paid services.<sup>17</sup> Their survey designs follow the same paged survey layout of the previous systems and do not incorporate any tablet specific controls. REDCap Survey, an offshoot of the REDCap project, provides another open system for survey collection that is appropriate for the medical setting. The two REDCap systems were originally developed separately, using individual databases that did not directly share data. However, the systems were linked in February 2011 to enhance data collection capabilities. It is currently one of the largest working systems, providing data capture tools for biomedical research to a group of 213 institutional partners<sup>2</sup>.

Leveraging modern systems improves researchers' capability to collect data while reducing data error and decreasing interviewers' time requirements<sup>8-10</sup>. For instance, allowing subjects to self-administer surveys, regardless of collection method, has been shown to be important for collecting sensitive data<sup>11,12</sup> as subjects have an increased sense of anonymity when they control their progress without direct assistance. Electronic systems for self-administration can also reduce errors by skipping questions irrelevant to subjects based on previous responses (i.e., skip/flow logic), thereby simplifying portions of the survey for subjects and ensuring that individuals do not answer sections that should be left unanswered. More importantly, real-time validation of responses can be performed to catch errors, requesting corrections from the subject on-the-fly. Such tools also have the potential to accelerate the process of screening, recruitment, and enrollment of subjects. For example, the NIDA QUIT study anticipates screening over 8,000 individuals to reach a target enrollment of 500 subjects. Manually administering paper forms to such a large group, and subsequently determining eligibility for the study is a daunting task. In contrast, a computer system can evaluate responses as they are provided, immediately ascertaining if a subject is eligible for further surveys and study enrollment. In addition, automated randomization becomes possible for the research team as the system has access to centralized data for all enrolled subjects. Utilizing automated email and Short Message Service (SMS) communications can also improve the retention of subjects for follow-up and longitudinal work.

Unfortunately, current systems are not well-suited for handling surveys in our collection environment of community health centers. When our project began, only the REDCap system provided tools for building structured surveys for data collection in addition to administrative clinical trials tools. However, the two systems (REDCap and REDCap Survey) were not connected when we started. Also, running the survey in a web browser presents many usability issues. Our target population often has limited exposure to computers and the Internet, which can limit their ability to navigate through pages in a browser layout. Features common to most users, like tabs, navigation buttons, and address bars may distract less experienced computer users. Similarly, the population has a low literacy rate and therefore incorporating text-to-speech (TTS) functionality is necessary. While general-purpose TTS browser plugins and software exist, such as WebAnywhere<sup>13</sup> and NonVisual Desktop Access<sup>14</sup>, they lack the flexibility and configurability to be smoothly integrated into a data capture system. Finally, a touch screen interface for data input is desirable for less experienced users and for portability purposes. Although web-based survey pages in other systems could be tailored to the touch screen, the results are often suboptimal when compared to the browser experience and input devices of a standard computer. These deficiencies have led us to the current work with EMMA, where we have built a survey system around data collection on tablet computers.

## System Framework

### Survey Model

EMMA surveys are modeled using eXtensible Markup Language (XML)<sup>15</sup>. XML was chosen due to its simplicity, flexibility, and human readability. However, a freeform XML file alone is not a rigid definition. An XML schema<sup>16</sup> is therefore used to impose the exact structure for a survey XML file. For example, a similar open-source XML schema, queXML, dictates the survey structures that drive LimeSurvey.<sup>18</sup> The EMMA schema was developed to address the specific needs for a tablet presentation with robust storage and description of questions, scores, and skip logic. Existing schemas were not able to address system needs for features such as multi-language support, question "gating," and simple survey reuse across future studies. Descriptions for key elements of the EMMA schema are provided in the following sections.

**Survey element.** The <Survey> element is the root of a Survey XML file. It consists of attributes like ID and Description that pertain to the survey at a global level. Each <Survey> element contains multiple <Slide> elements, which correspond to the questions in the survey. An extremely flexible design for the <Slide> element was chosen to enable it to model different question types using its children elements, <Question>, <Input> and <Next>.

The <Question> element specifies the question text that is presented to a user. Each <Question> contains multiple <Context> elements, each containing the question text in a different supported language. This structure thus allows additional languages to be incorporated with minimal redundancy. The <Question> tag also supports a calendar display by including a <Calendar> element. This tag is particularly useful for answering questions related to dates. A date range can be optionally highlighted on the calendar by specifying the StartDate and EndDate attributes of <Calendar>. The <Input> and <Next> elements are more complex elements for describing the answer set and flow-logic of the <Slide> and are elaborated in subsequent sections.

The <Slide> element provides a number of configurable attributes to facilitate data collection. For example, by setting the Tentative attribute, the user's response to the question is stored temporarily in memory but not in the final database. This feature is useful for responses containing sensitive information that should not be stored, but will be used to dictate skip logic later in the survey. Another important attribute is provided for "gating." When set, the user is blocked from returning to any questions that precede the current question; this protects previously collected data at major milestones (e.g., at the end of a given process or following randomization). Gating also blocks subjects who attempt to "game" the system by learning all the different response combinations necessary to avoid exclusion (to gain the monetary incentive given for participation).

**Table 1.** Question types supported using the <Input> element of the XML survey model

Input Type	Description
Nothing	Serves as a placeholder for information-only questions. The information itself takes up the entire screen, and the only response expected from the users is to proceed to the next question.
Multiple Choice	Options are modeled by the <Choice> element. Allowance for single or multiple responses is set in the attributes. Additional configurable parameters include layout style, exclusive response (e.g., "None of the above"), and allowance for user-entered information (e.g., "Other, please specify").
Matrix	A collection of multiple-choice questions that share the same choice options. On paper, this is often shown in a tabular form. A common example is Likert-scale questions.
Date	A simple month-day-year mini-wizard is shown to guide the user through the date entering process. MaxDate and MinDate attributes can validate the range of dates that the user is allowed to input.
Number	Used for entering numeric responses. A standard number pad is displayed on the screen for the user. The maximum and minimum allowable value can be specified using the Max and Min attributes.
Free Text	Allows the user to enter short or long textual responses. An onscreen keyboard is provided for users to type their response.
Consent Signature	Input type for obtaining user's consent. A dedicated area is displayed on the screen for the user to sign using a stylus. The signature is captured as a bitmap file.

**Input element.** Each <Input> element in a <Slide> encompasses one of seven supported input type elements. These types describe the possible answer cases available for the question asked. The design allows future question types to be supported with minimal modifications. Similar to the <Question> element, localization is supported wherever text is displayed. Supported input types and the matching elements are listed in Table 1.

**Next elements.** EMMA provides a highly configurable logic system. All logic testing can be constructed based on the five different sources presented in Table 2. The system is used for both controlling skip patterns and score computations, which are modeled by the <Next> and <Score> children elements of <Slide>.

**Table 2.** Logic types supported in the <Next> element of the XML survey model

Logic Type	Description
Answer-based	Logic testing can be performed based on the user's responses to preceding questions. Possible testing includes min/max value, substring matching, and the selection of a particular choice.
Score-based	Score-based logics are evaluated based on the value of a particular score. The testing range can be specified using the MaxScore and MinScore attributes.
Visit-based	Visit-based logic testing can be evaluated based on whether a particular question has been seen by the user previously.
Web-based	Requests logic testing using an evaluation performed at the server-end on stored data. Targets a server URL, and the parameters to be posted can be customized depending on logic requirements.
Script-based	A script-based logic provides the ultimate configurability by using an embedded Python script. This allows customized logic testing based on nearly every possible permutation.

**Score element.** Apart from <Slide> elements, each <Survey> can also contain multiple <Score> elements. Each <Score> corresponds to a particular score to be tracked for the duration of the survey and is denoted using an ID and textual name. The maximum and minimum value of the score can be capped using the MaxScore and MinScore attributes.

**Process elements.** A complicated data collection session may require a set of surveys to be filled out as a group. The <Process> element joins multiple surveys into a single unit and stores the order that they are presented to a subject. For example, the screening, consent, and enrollment steps are divided into separate surveys in the QUIT study. The process feature provides a method to join such contiguous steps into a single survey flow, while maintaining their independence for possible use in other processes. Parallel to skip patterns, the sequence of surveys may be controlled using <FlowLogic> elements. All previously described logic types can be used by a process, but only the transition between studies is affected. The <Randomization> element, specific to processes, is a special extension to the web-based logic, where the outcome can be *n*-ary instead of binary. Finally, a <Process> also supports global string substitution through <Variable> elements. This is particularly useful when the same bundle of surveys is reused with minor name changes, for example to change the collection site name.

#### *Data Collection – Tablet Software*

The replacement for paper surveys is a graphical user interface (GUI) written in C# .NET, providing access to designed surveys and the visual interpretation of the stored XML survey models. This software supports netbook computers running a version of Windows XP (or newer). The NIDA QUIT study, the first to use this system, purchased ten Asus Eee PC T91 netbooks for data collection. The T91 is a convertible tablet PC with an 8.9" resistive touch screen. Internet in our target clinics is largely unavailable due to infrastructure or security issues. To establish access to the database, wireless mobile hotspots were purchased from Verizon, providing 3G connectivity for the tablets. The hotspots are configured to provide secure access only to our computers to protect any information transmitted to and from the database.

Rather than provide a web-based survey viewed in a browser, as done in previous methods, our system translates the saved XML survey model description into a series of individual questions. A motivating reason for this adaptation is the reduced visual space available on the tablet screen. Under this constraint, long surveys are difficult to render and can require a significant amount of scrolling for navigation. Figure 1 demonstrates how questions are rendered by the software to optimize the use of screen space: the GUI attempts to maximize all on-screen text and buttons to make the survey easy to read. Breaking the survey into a set of individual questions is also advantageous as it reduc-

es the users' cognitive load. Reading questions using TTS is simplified and the text being presented is obvious to the user, as only a single question and answer set are visible and dictated. This approach ensures that respondents focus on the question at hand, and avoids overwhelming subjects with multiple questions simultaneously. Notably, skipping irrelevant questions (based on user responses) becomes a transparent process: users are unaware of "skipped" questions. These design decisions to reduce graphical complexity were an appealing aspect to the NIDA QUIT study team given the anticipated issues of literacy limitations in the subject population.



**Figure 1.** Input types as rendered by tablet software. From left to right: Information, Multiple Choice, Date, Free text, Signature.

System use begins when an interviewer logs in and is presented with a list of studies at the institution. After a study is selected, the system displays the available processes that may be used under the selection. When interviewing new subjects, a process is selected, generating a new unique ID for the interviewee and a set of screening surveys. Following ID creation, interviewers proceed by administering questions verbally in a formal interview, or by handing the tablet over to the subject for self-administration. When conducting a follow-up with past subjects, an interviewer provides the previously generated subject ID at the start of a survey to link newly collected data to the subject's existing record.

As subjects can be excluded by study criteria or may have to withdraw from the study at any time, methods are provided to record these occurrences. To track exclusion criteria based on subject responses, designers are required to include additional questions that capture the exclusion event and document the reason for exclusion. If subjects withdraw prior to completing a survey, an interviewer interrupts data collection early with a preset pin code and selects the reason for withdrawal from a defined list of codes (and supplies further text explanation when necessary). These values are stored as additional metadata and appended to the survey data.

Survey responses are serialized and stored locally following each question response in order to protect against power failure, system crash, or network disruptions. Simple status event data is also sent to the server following responses. These events generate a report for coordinators to track subjects' progress in real-time. Tracking provides time-stamps for each tablet, which helps to indicate when a subject may be stuck on a section of the survey and is not asking for assistance. Interviewers are able to pause sessions (e.g., because of an interrupted session that needs to be continued at a future time point), in which case the tablet software sends a copy of the data to the server for storage and removes any local copies. Additional controls also allow interviewers to skip questions if subjects have objections to answering. Survey data is immediately sent to the central database for storage at the completion of a process and all local copies are deleted.

#### Web-based Survey Designer

A survey designer that is simple to use by non-programmers, while providing access to sophisticated control logic, was found to be a necessary element for EMMA acceptance. EMMA employs a web-based interface to design all surveys and processes to be used with the tablets for data collection. The interface is built in ASP.NET and has controls for data administration (discussed below). As mentioned previously, EMMA's survey model consists of two top level constructs: *processes* and *surveys*. Designers make use of processes to group smaller question sets into a

full survey or sets of surveys into large survey sets. This abstraction encourages the deconstruction of large studies into logical sections to promote reuse (either within the same study or future studies). The NIDA QUIT study, for instance, has a survey designed to ask questions pulled from the World Health Organizations (WHO) Alcohol, Smoking and Substance Involvement Screening Test (ASSIST). The WHO ASSIST survey is of particular importance for testing if a subject has developed risky use or dependence on a variety of substances (drugs, tobacco, and alcohol). Knowing that the WHO ASSIST questions will be used for many other studies performed by the UCLA Family Medicine team, a separate survey was created to expedite reuse in future processes.

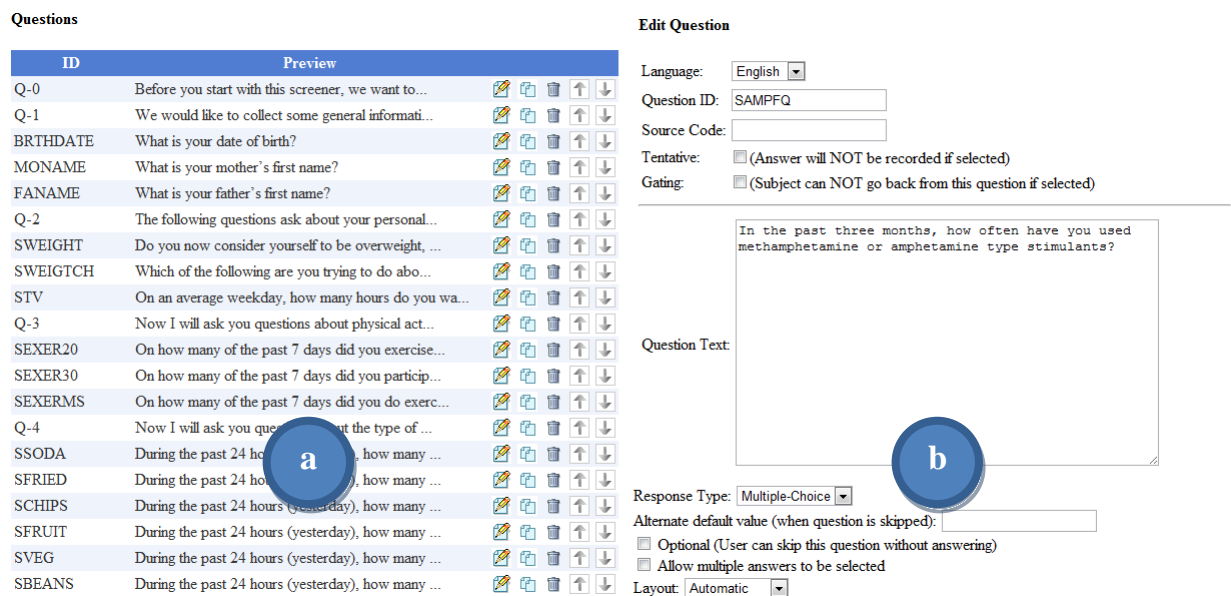
A user's primary goal when designing a survey is to first establish the questions that will be asked of subjects. The survey designer allows the user to choose from the several question response types described in our model: information, multiple choice, matrix, dates, numbers, free-text, and signatures. Each type is similar to those provided in other web-based systems, with some additional capabilities. For example, the free-text question type allows the designer to include a comma separated list of words or phrases to be used for auto-completion for the user. As questions are designed, studies requiring multi-language support (e.g., Spanish) can provide a translated version of the questions and responses. Basic scripting can be added to the text of any question to allow programmatic inclusion of phrases or past data values. This ability is commonly used in the QUIT study for questions that reference only the drugs determined to be used by responses to previous questions (i.e., if a respondent confirms they use only cocaine, future questions will refer only to cocaine, rather than the complete list of possible drugs). A similar use case is filling in dates in questions. The *Date* script allows the designer to write a question asking about a date *n* weeks in the past, but have that date explicitly computed in the text based on the current date to remind the subject.

After establishing the set of questions to be asked, the second goal for a study design user is to establish navigation and logical flow through the survey. The EMMA survey designer allows questions to be flexibly reordered as needed by attaching a number of different programmatic elements:

1. **Skip patterns.** Logical constraints can be associated with each question to control question presentation; these rules are referred to as *skip patterns*. A skip pattern's logical conditions are evaluated following a response to a question, and allows the designer to jump to another question in the survey. Available skip pattern criteria were previously described in the model description.
2. **Scoring logic.** Separate from question logic, user responses can be used to update score variables for tabulating numeric values over the life of the process. Scoring logic follows the same design structure as skip logic, but only allows the system to update the value of a stored score variable.

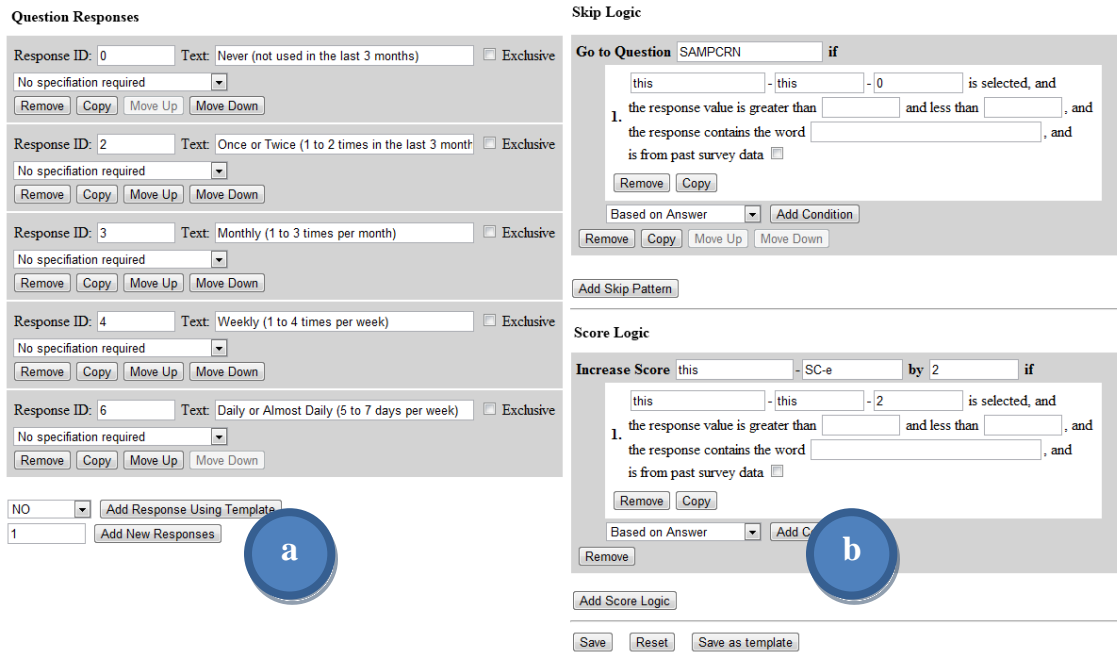
Common question layouts can be saved to a bank of templates for reuse in future survey development. The survey designer layout and some common tools available to survey designers are shown in Figure 2 and Figure 3.

Study coordinators designate the top level processes using the survey designer module. Processes must link to at least one survey to be valid but can group several surveys to be run together. Skip logic must be designated to con-



**Figure 2.** Survey designer pages displaying a) the listing of current survey questions, b) controls for editing the question text.

trol the flow when multiple surveys are included. For follow-up surveys, past data can be requested for inclusion to aid in question scripting and skip patterns; this data is packaged for download from the server when the process is started on the tablet. Finally, process settings control the timing and web request instructions to perform randomization if needed in a study.



**Figure 3.** Survey designer pages displaying a) controls for editing the question responses, and b) controls for designating skip and score logic for the question.

*Web-based Administration*

EMMA’s web-based system facilitates a variety of study, subject, and data management operations. These administrative tools are divided into sets of web pages appropriate to three divisions of management covered in Table 3. Important pages for each module are described further below.

**Table 3.** Management modules for administrative web site tasks.

Website Module	Description
Study management	Contains web pages for setting study variables and settings, tablet tracking reports, staff scheduling, and the previously described Survey Designer.
Subject management	Contains web pages supporting daily management of subject data. Individual pages cover collected survey data, demographics, outside data (e.g. lab reports), adverse event tracking, future scheduling, and records of follow-up attempts.
Data management	Contains web pages for generated data reports and tools for data export for statistical analysis.

**Study Management.** Global settings for each study are particularly important for administrators to establish prior to collection. The study settings page groups the following important variables and settings:

1. **Included processes.** The given list of processes the interviewers can access with the tablet software.
2. **Site locations.** The list of remote sites, such as community health centers, involved in the data collection process. Administrators describe site names and locations which allows the tablet to automatically provide a list to available interviewers at each site. Site selection provides unique identification in the system for future analysis of where data was collected.
3. **Monetary incentives.** A description of the types of incentives available to subjects and the amounts to be paid out. Tracking forms for incentives are then automatically generated in other subject management pages.

4. Study stages. A description of data collection stages and their order in the study. Common examples of stage markers are screening and recruitment, enrollment, monthly follow-ups, intervention activities, and completion of collection.
5. User permissions. Access controls designating the appropriate viewable content for interviewers, health educators, and any additional system users.

Subject Management. All subjects observed, screened or enrolled using the tablet during data collection are listed in a main table view. Study coordinators can review the listing of subjects and access pages for editing and adding demographic information and for scheduling follow-up events. If stage information is provided, coordinators also receive a quick visual flag describing the days remaining to schedule a follow-up. If permissions allow, a user can review the data stored in the database for subjects individually. Coordinators can also import external records such as the subject's medical record or lab data as needed. Adverse events reporting for the study is also included in order to track problems for reporting to an institutional review board (IRB). To follow-up with subjects, a second scheduling system indicates upcoming phone calls with subjects only. Records are also generated describing the result of attempted contact for scheduled calls.

Data Management. Study- and system-wide reports are provided with summaries of the database and its' contents available to coordinators; this information is needed by study administrators on a regular basis. For example, a report of monthly screening and enrollment statistics is provided to track the progress of the study. Another report focuses on the current status of all subjects in the study based on the stages designated in the global study variables. This allows coordinators to track what surveys are completed and what follow-ups remain. Reports also exist to display randomization counts, subjects who have earned monetary incentives and need payment, and a tracking report for which interviewers are assisting subjects. It is not currently possible for the coordinators to design their own reports. Thus, these reports were developed with input from the NIDA QUIT team. In order to support full data analysis, study coordinators can export survey results for all collected patients into a comma separated value file for import into Excel or other statistical packages.

## **Discussion**

The presented system was developed over the last year. Data collection for the NIDA QUIT pilot study has been running for six months. In that time, the system has been run by a team of 3 coordinators and 25 volunteer research assistants. Initial feedback on system use has been positive. Coordinators and volunteers in the QUIT study report the tablet software easy to use during the approach process and report few cases where subjects require training to understand and use the system. The QUIT study has thus far screened 602 subjects over the last six months, enrolling 81 into the trial for treatment and follow-up. Current discussions include expansion of system use at additional clinics, the creation of new surveys for ongoing studies within UCLA Family Medicine, and changes necessary to provide the system to other institutions. Over the course of development, our work has also noted potential complications and pitfalls for survey systems that must be addressed by programmers and coordinators.

The generalized survey model is robust to many design strategies. However, survey designs exhibited an unexpected complexity and need for control of certain features. Skip logic, for instance, is a feature designers do not need to consider for paper-based surveying. Although simple logic checks can define the ordering of most process and survey conditions, designers quickly indicated there were scenarios that required more complex control. For example, initially the scope of skip logic conditions was limited to variables within a process. However, designers also wished to incorporate data collected in surveys from earlier processes into skip logic rules. To resolve this issue, additional abstracted functions were developed for survey designers to embed such information as part of skip logic constraints. Other complex logic tasks drove the inclusion of the scripting logic type for our system to cover as many designs as possible; though this compromise necessitates additional training for future survey designers.

Programmers must remain cognizant of the available resources that dictate tablet processing speed and network connectivity. The Asus EeePC T91 chosen for this project is equipped with limited processor and memory capabilities. Coding was optimized over a number of test periods to provide a smooth user experience. Likewise, system design had to account for limitations in 3G wireless connectivity at the clinic sites. Issues with data transfer speeds and signal strength required compromises in the number of requests sent to the database. Data required by a process is queried and cached during survey loading rather than sending individual requests to avoid unnecessary delays during subject use. Similarly, the system constantly backs up user responses to the local computer and complete questions are sent to the server in large batches to protect against connection interruptions.

Future work will consider the evolving market for tablet computing. At the start of this project, the iOS and Android systems were restricted primarily to phones. Both have since adapted to tablet formats and are proving their efficacy.



These new systems offer improvements such as built in keyboard and alternate data input methods, text completion and correction, and longer battery life that will encourage further adoption of tablet use in the field. Development of a web-based EMMA client for these systems is planned to make the system available on a variety of tablet systems for future use at outside institutions.

### Acknowledgements

This research was supported in part by the NLM Training Grant LM007356 and the NIDA R01 DA 022445.

### References

1. Couper MP. Technology Trends in Survey Data Collection. *Social Science Computer Review* 2005 Winter;23(4):486 -501.
2. Harris PA, Taylor R, Thielke R, Payne J, Gonzalez N, Conde JG. Research electronic data capture (REDCap)-- A metadata-driven methodology and workflow process for providing translational research informatics support. *Journal of Biomedical Informatics* 2009 Apr;42(2):377-381.
3. REDCap. <http://project-redcap.org/>. Last accessed March 17, 2011.
4. Trial DB. <http://ycmi.med.yale.edu/TrialDB/>. Last accessed March 17, 2011.
5. OpenClinica. Open source for clinical research. <http://www.openclinica.org/>. Last accessed March 17, 2011.
6. Survey Monkey. <http://www.surveymonkey.com/>. Last accessed March 17, 2011.
7. SurveyGizmo. <http://www.surveygizmo.com/>. Last accessed March 17, 2011.
8. Richter JG, Becker A, Koch T, Nixdorf M, Willers R, Monser R, Schacher B, Alten R, Specker C, Schneider M. Self-assessments of patients via Tablet PC in routine patient care: comparison with standardised paper questionnaires. *Ann. Rheum. Dis* 2008 Dec;67(12):1739-1741.
9. Gwaltney CJ, Shields AL, Shiffman S. Equivalence of electronic and paper-and-pencil administration of patient-reported outcome measures: a meta-analytic review. *Value Health* 2008 Apr;11(2):322-333.
10. Weber BA, Yarandi H, Rowe MA, Weber JP. A comparison study: paper-based versus web-based data collection and management. *Applied Nursing Research* 2005 Aug;18(3):182-185.
11. Bernabe-Ortiz A, Curioso W, Gonzales M, Evangelista W, Castagnetto J, Carcamo C, Hughes J, Garcia P, Garnett G, Holmes K. Handheld computers for self-administered sensitive data collection: A comparative study in Peru. *BMC Medical Informatics and Decision Making* 2008;8(1):11.
12. Chang L, Krosnick JA. Comparing Oral Interviewing with Self-Administered Computerized Questionnaires An Experiment. *Public Opinion Quarterly* 2010 Mar;74(1):154 -167.
13. WebAnywhere. <http://webanywhere.cs.washington.edu/>. Last accessed March 17, 2011.
14. NonVisual Desktop Access. <http://www.nvda-project.org/>. Last accessed March 17, 2011.
15. eXtensible Markup Language (XML). <http://www.w3.org/XML/>. Last accessed March 17, 2011.
16. XML Schema 1.1. <http://www.w3.org/XML/Schema>. Last accessed March 17, 2011.
17. Lime Survey. <http://www.limesurvey.org/>. Last accessed July 13, 2011.
18. queXML. <http://quexml.sourceforge.net/>. Last accessed July 13, 2011.