7-20-2004

An Online Survey Framework Using the Life Events Calendar

Jared Kite
University of Nebraska, kite@cse.unl.edu

Leen-Kiat Soh
University of Nebraska, lsoh2@unl.edu

University of Nebraska–Lincoln, Computer Science and Engineering
Technical Report TR-UNL-CSE-2004-0011
Issued 7/20/2004

An Online Survey Framework Using the Life Events Calendar

Jared Kite and Dr. Leen-Kiat Soh
(kite@cse.unl.edu, lksoh@cse.unl.edu)
Computer Science and Engineering
University of Nebraska–Lincoln
Lincoln, NE 66588-0115

July 20, 2004

Abstract

We describe an online survey framework programmed as a Java applet with a MySQL back-end. Our framework is built specifically as a Event History Calendar for the study of tobacco users and their behavior over a six month period. We introduce the notion of a Life Events Calendar and the relevance of an intelligent survey system in this context. We describe our methods and our component application approach and expand on the opportunities for artificial intelligence research with the system.

Contents

1 Introduction

Event History Calendars are survey instruments that are well suited for collecting retrospective autobiographical information. However, these instruments introduce a more conversational method of interview, and expert interviewers are necessary to guide the subject through the process. Artificial intelligence techniques have potential for improving the ability of the computer to help in the interview process, removing the need for expensive human experts. Some artificial intelligence research has been done with instruments related to Event History Calendars but little has been with Event History Calendars themselves.

We create an online Event History Calendar for the study of tobacco users and their habits over a six month period. We present our custom built system and plan for integrating artificial intelligence techniques to improve the ability of the computer to assist in the interview process. We discuss work related to ours, the design and implementation of our system and then the possibilities for adding artificial intelligence components.

2 Background and Related Work

2.1 Event History Calendars

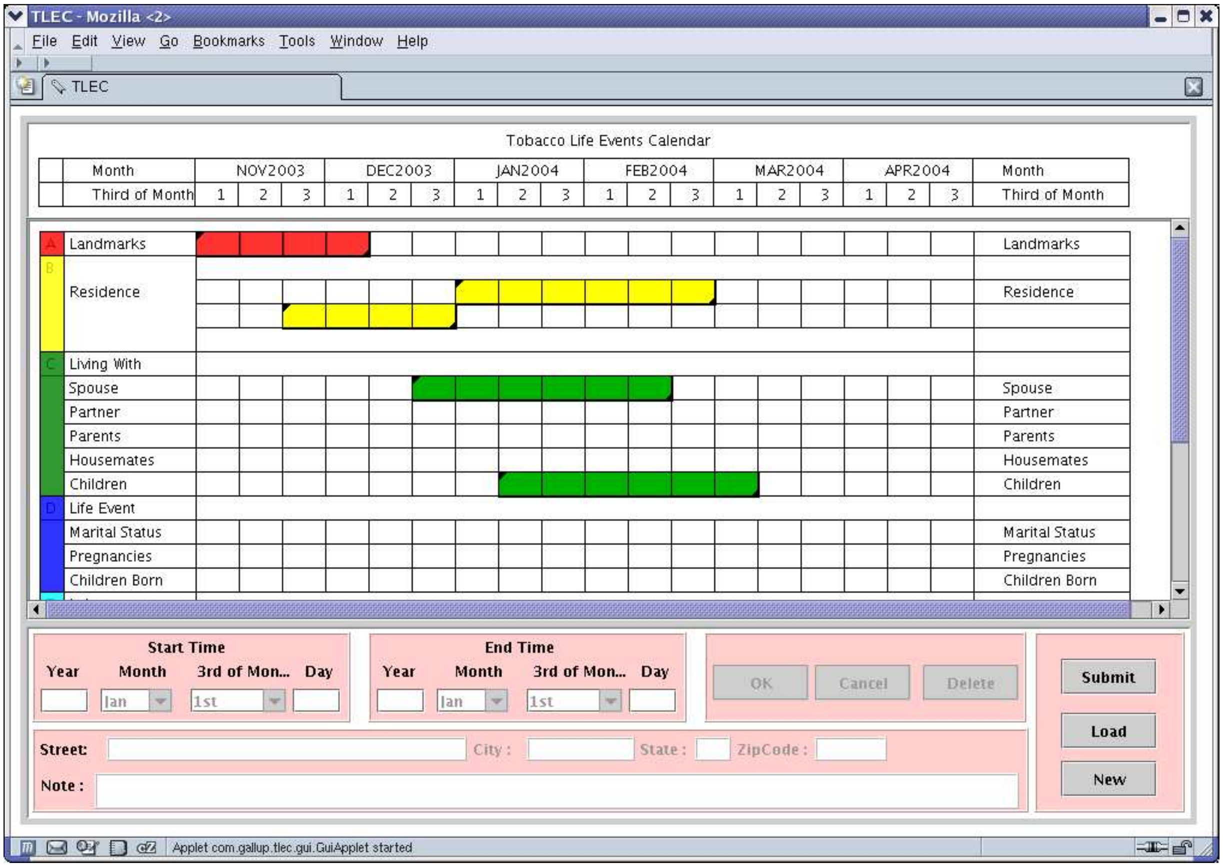
Event history calendars are an alternative to the standardized question list type survey. Event History Calendars were first described as Life History Calendars in 1988 by Freedman et al. [5]. We compare question list surveys to Event History Calendars in order to illustrate the characteristics of this relatively new survey type.

A question list survey is the most popular survey type and rightfully so according to current survey methodology [4, 7]. A question list survey is a list of carefully selected questions that are organized by topic. The possible answers to each question are usually limited by closed-ended questions or questions that are fact based [3]. The closed-ended nature of the questions is designed to reduce error in the survey results.

Question lists are good for certain surveys that probe attitudes, but not for tasks where the goal is gaining retrospective data [3]. When it is desirable for the subject to accurately recall events that happened years before, event history calendars are the better choice [3]. This is particularly true when the events are autobiographical, or part of the the subject's own life history. An Event History Calendar is structured to take advantage of the ways that autobiographical memories are stored and structured [1].

The nature of an Event History Calendar more closely matches the human model for autobiographical memory than does a question list [1]. The style of interviewing that the Event History Calendar encourages is narrative in style in which the remembering of events from an individual's past act as cues for the recall of other related events [2]. There are three cueing mechanisms provided by an Event History Calendar that promote memory recall: top-down, sequential and parallel [2]. The structure of autobiographical memory is composed of lifetime periods, general events and specific events organized into a hierarchical structure with lifetime periods at the top [1]. Each of the three cueing mechanisms relate to a separate way that autobiographical memories are structured [1].

The remembering process travels down the hierarchy of memory. Top-down cueing occurs


TLEC - Mozilla <2>
File Edit View Go Bookmarks Tools Window Help
TLEC
Applet com.gallup.tlec.gui.GuiApplet started

Tobacco Life Events Calendar

		NOV2003			DEC2003			JAN2004			FEB2004			MAR2004			APR2004			
	Month	NOV2003			DEC2003			JAN2004			FEB2004			MAR2004			APR2004			Month
	Third of Month	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	Third of Month
A	Landmarks	🟥	🟥	🟥	🟥															Landmarks
B																				
	Residence							🟨	🟨	🟨	🟨	🟨	🟨							Residence
				🟨	🟨	🟨	🟨													
C	Living With																			
	Spouse						🟩	🟩	🟩	🟩	🟩	🟩								Spouse
	Partner																			Partner
	Parents																			Parents
	Housemates																			Housemates
	Children								🟩	🟩	🟩	🟩	🟩	🟩						Children
D	Life Event																			
	Marital Status																			Marital Status
	Pregnancies																			Pregnancies
	Children Born																			Children Born

Start Time				End Time			
Year	Month	3rd of Mon...	Day	Year	Month	3rd of Mon...	Day
	Jan	1st			Jan	1st	

OK Cancel Delete

Submit

Load

New

Street: City : State : ZipCode :

Note :

Figure 1: This is our Event History Calendar applet interface. This shows the entire applet running inside of the Mozilla web browser.

when memories of lifetime periods bring about memories of general events which then spark memories of specific events [2]. Sequential cueing refers to the recall of general or specific events that occurred in a chronological fashion inside of a lifetime period [2]. For example, the memory of a leaving one job may spark the memory of beginning another. The third, parallel cueing, deals with memories of events that are contemporaneous and occurring across lifetime periods [2]. For example, changes in one's job or personal relationships may be related to how much they smoke.

A paper and pencil Event History Calendar allows the subject to record their lifetime periods and events that span time on a structured form. The form has domains (major events, marriage, divorce, childbirth, etc...) intersected with a time scale in a grid format. There are two main types of Event History Calendar layouts, horizontal and vertical. In the horizontal case, the time scale is laid out horizontally and the topics vertically. The vertical case is the exact opposite. An example of our Event History Calendar, a horizontal one, is shown in Figure 1. Completing the survey consists recording the subject's autobiographical memories in the proper categories with as much veracity as possible.

There are distinct uses for an Event History Calendar survey instrument. An Event History Calendar may be used as a companion to a question list survey by aiding autobiographical memory recall. An Event History Calendar may also be used as the survey instrument itself. In either of these situations, the instruments may be paper and pencil based or computerized.

2.2 Computer Assisted Interview

Computer assisted interview (CAI) uses a computer in some capacity for survey administration. Generally, with Event History Calendars, computerizing a survey leads to decreased time spent transcribing data and more complete and consistent data. The computerization of the survey can include functions that assist the survey administrator in recognizing inconsistencies in the data as well as assist the interviewer in recognizing good situations in which to probe the subject's memory using cues. This can resolve inconsistencies and increase memory recall. The disadvantages to CAI are mainly those commonly associated with using computers in the field including electricity, availability, and subject familiarity with computers.

2.3 Previous Efforts with Electronic Event History Calendars

In 2000 Wiebe and Landis built two computerized Event History Calendars, one each of horizontal and vertical, for the purpose of comparing the two [10]. The points of comparison were mainly the "accuracy and accessibility of the information recorded in the Event History Calendar" [10]. Based upon user evaluations and tests they found that the horizontal Event History Calendar was, overall, better for ensuring data accuracy [10]. They also discovered that the horizontal Event History Calendar was slightly preferred by the users [10]. They did a detailed study of each respondent to their study. However, their sample size is small and they suggest a large scale study to verify their results [10].

A computerized Event History Calendar similar to ours was created at the University of Michigan by Belli [2]. Belli's study primarily examined the the switch from paper Event History Calendars to computerized Event History Calendars and the graphical design of the Event History Calendar. He notes that the disadvantages of using a paper Event History Calendar include significant time investments for transcription of the data to computer files, and the failure of the interviewers to ask valuable questions. He proposes that a computerized version of the Event History Calendar might alleviate these problems. Along with reducing the time of transcription, they have the potential to improve subject autobiographical recall by notifying the interviewer in situations when a probe is useful [2].

The application described by Belli was designed for the Microsoft Windows operating system using Microsoft Visual Basic. It was supported by by a Microsoft Access database for all data storage.

The application represents hierarchical organization with tabs across the top, one for each domain. Each domain has its own time-line entry screen. During the interview process, inconsistencies and gaps in the data provided are checked. When a problem is found, it is highlighted for the interviewer. At this point, the interviewer can use an appropriate probe to resolve the problem.

There are almost no documented attempts to implement a computerized Event History Calendar that utilizes artificial intelligence techniques. The idea of an Event History Calendar is quite new, so the lack of enough experts and testing could explain their limited adoption up to this point. Additionally, those that design these systems are not Computer Science researchers. Consequently, they are technology adopters rather than technology developers. This has lead to the somewhat slow integration of Computer Science techniques that could be very helpful to Event History Calendar surveys. Basic checks, such as those in [2] do assist the interviewer. However, we can apply Artificial Intelligence to the problem and increase the power of the computer to assist.

2.4 Previous Efforts with Related Instruments

Although there have not been documented cases of Computer Science techniques applied to Event History Calendar instruments they have been applied to other types of instruments. An example, applied to computer aided interviewing, is given by Poulsen [9]. He applies artificial intelligence techniques to question list surveys to segment the subjects into groups and to minimize the number of questions asked. He uses Adaptive Bayes classification to choose the next question that will increase the error the least [9].

Segmenting the subjects into predefined groups, also called coding, is a popular technique in marketing research where people can be separated into groups related to product interest. A general study of automated survey coding as a text classification problem was presented in 2003 by Giorgetti and Sebastiani [6]. They use naive Bayesian and Support Vector Machine classifiers to segment the subjects into distinct groups. Their results outperform previous efforts in automated coding [6]. This is a technique that could possibly be applied to an Event History Calendar survey instrument.

3 Implementation

Our implementation includes an electronic and online Event History Calendar specified by an expert in the field. It will be used as a data collection instrument in a Gallup survey of tobacco users and their habits. The application consists of a GUI front-end along with a database for all data storage. We also plan for an intelligent survey assistant module that will enhance the usefulness of the instrument. Figure 2 gives a general idea of the application organization.

3.1 Event History Calendar Design

Our Tobacco Life Events Calendar Event History Calendar is implemented as a Java applet and tested with Microsoft Internet Explorer on the Windows XP operating system. The applet can be used with any Java enabled browser on any operating system that supports such a browser. Because Java applets are Web-based, it is available on the World Wide Web to anyone who has Internet access. This is a major advantage; we can support a wide variety of survey situations.

The design of our Event History Calendar is specified by Dr. Bob Belli, a leading researcher in the area of Event History Calendar use. Our Calendar stores information on tobacco users and their habits six months previous to the survey date. The Event History Calendar is the sole data collection instrument for our Tobacco Life Events project.

As shown in Figure 1, our Event History Calendar is horizontal. A number of features are included to enhance the three types of memory recall: top-down, parallel and sequential. The time scale remains stationary while the user scrolls through the rows of domains to speed event entry to aid in sequential cueing. The domains are clearly marked with colors and each specific event is labeled on both sides of the screen to aid in top-down processing. Notice that the Landmark and

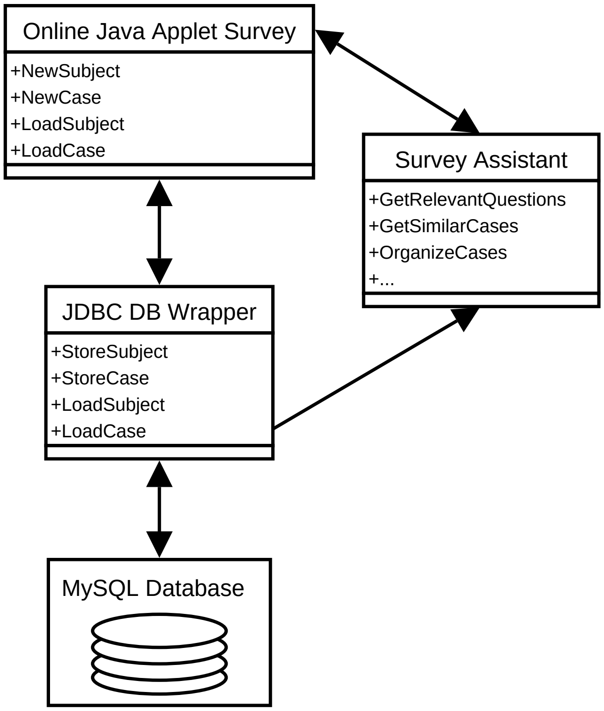


Figure 2: This is the general organization of the application. This shows the major components of the application and their basic functionality.

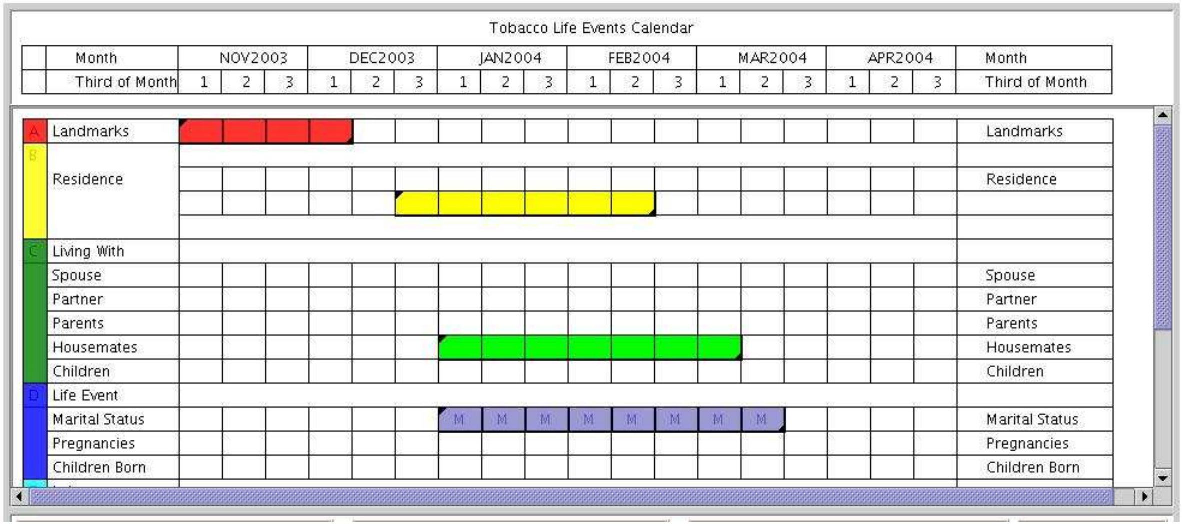
Residence events are positioned at the top and when filled out first aid in top-down processing for all of the domains. Parallel processing is aided with the color coding of the events and the fact that related topics are above and below each other as they are in autobiographical memory. Also notice that the beginning and end of every event entered is marked with small triangles in the corner of the beginning and ending rectangles, shown in Figure 1. When an event is added to the Marriage Status category, letters are shown in all of the boxes indicate the marital status and act as both parallel and sequential cues. There is also evidence that using landmark events, as we do, will help to establish a solid time-line for the subject and increase the accuracy of their autobiographical memories [8].

There are two main graphical components to our implementation, the Event History Calendar display on top and the data entry panel on the bottom. An example of the Event History Calendar display is shown in Figure 3 and the data entry panel in Figure 4. An event is entered by clicking and dragging the mouse from the beginning of the event to the end on the Event History Calendar display. The user can enter the details about the event and save their work. When finished adding events, the user may submit the entire survey to the database.

We now look at the specific data that is collected for each domain in the Event History Calendar. For every event entered in the calendar, a begin date and an end date are recorded along with a text annotation field for notes. Beyond those items, each domain has unique fields to be filled out.

Landmark Events: This section requires a description of the event.

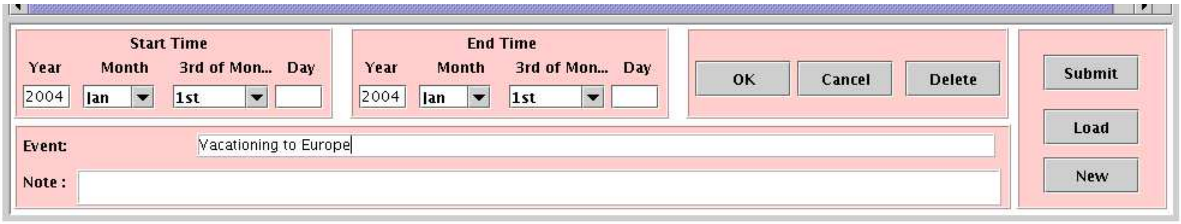
Residence: This section requires the address of the subject's residence



Tobacco Life Events Calendar

	Month	NOV2003			DEC2003			JAN2004			FEB2004			MAR2004			APR2004			Month
	Third of Month	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	Third of Month
A	Landmarks	🟥	🟥	🟥	🟥															Landmarks
B	Residence																			Residence
							🟨	🟨	🟨	🟨	🟨	🟨								
C	Living With																			
	Spouse																			Spouse
	Partner																			Partner
	Parents																			Parents
	Housemates							🟩	🟩	🟩	🟩	🟩	🟩	🟩						Housemates
	Children																			Children
D	Life Event																			
	Marital Status							M	M	M	M	M	M	M	M					Marital Status
	Pregnancies																			Pregnancies
	Children Born																			Children Born

Figure 3: This is the calendar part of the Event History Calendar where events are entered and organized. This shows an example where four events are already entered. Notice the cues in each rectangle of the marital status event that indicate the subject's marital status.



Start Time

Year: 2004 Month: Jan 3rd of Mon...: 1st Day:

End Time

Year: 2004 Month: Jan 3rd of Mon...: 1st Day:

Event: Vacationing to Europe

Note :

Figure 4: This is the data entry portion of the Event History Calendar GUI. This shows an example of the data entry for a Landmark event.

Living With: This section requires the name(s) of the people that the subject was living with: spouse, partner(s), parent(s), house-mate(s), and/or child(ren)

Life Events: This includes descriptions of major life events. In the Marital Status category there is a choice of Married, Separated, Divorced, Widowed or Single. In the categories of pregnancies the woman's name is required and a child's name in Children Born.

Labor: This section requires the name(s) of the subject's school, main employer and/or position

Smoking: This section requires information about the subject's smoking habits. For the Smoke category, the choices are Everyday, Somedays, or Never. Amount per day asks for the number of packs smoked per day, and the choices are 0, 1-10, 11-20, or 20+. Awakening asks how long after awaking, in minutes, the subject began to smoke, with options of 1-10, 11-30, and 30+. The Quitting category asks what method the subject was using to quit smoking, if any. The options are not specified at the time of the writing of this report.

3.2 Java Application Design

The application is programmed in Java with the components shown in Figure 2. The classes with their relationships are shown in Figure 9. The main data structures in the application are stored in the Case and the Subject classes. These hold all of the data that is needed to keep track of every person and interview. Each event is stored in its own Event object. All of the Events associated with any particular case are placed together in a single Case object. A single Subject object can be associated with zero or more Case objects in the database, allowing a person to be interviewed multiple times.

Three major panels make up the GUI Applet along with an instance of the Case class storing the current case data. The data in the data structures is translated and displayed on the panels. As well as displaying the survey data, these panels are responsible for all of the GUI functionality of the application.

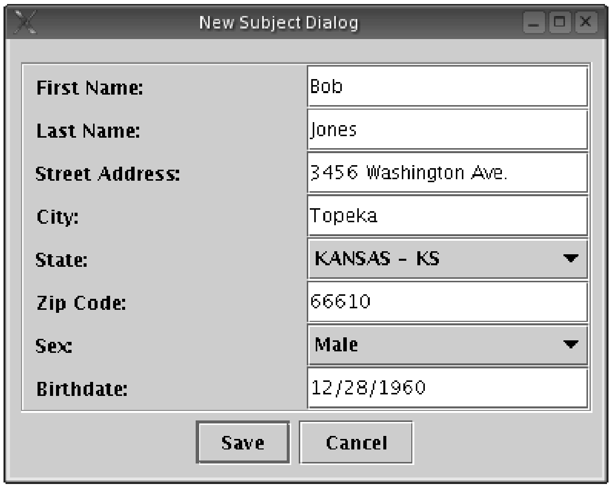
The flow of the application, from start to finish, is given in Figure 8. Each box in the diagram shows a point in the process of using the application. If the subject being interviewed is in the database, then it is chosen from the list of existing subjects shown in Figure 6. Figure 5 shows the dialog box used to enter the data for a new subject.

If a new subject is added, then a new case is created automatically and the user is sent directly to the edit phase. If the subject is chosen from the list of those existing, then the cases associated with that subject are shown in a dialog, Figure 7. The user chooses a case and is then sent to the edit phase.

In the edit phase, the user can fully edit a case. This includes adding, removing and editing existing events in the case. When the user is finished editing the entire case, it may be submitted to the database. From the edit phase, the user can either create a new case or edit an existing case for the current subject. Our flow allows for pausing and resuming an interview at a later time.

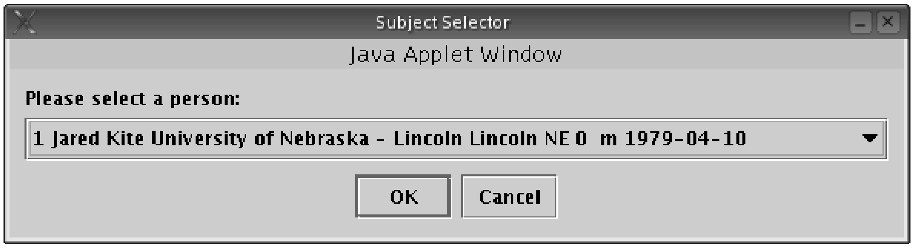
3.3 Database Design

The MySQL database system is used for storage of the survey data. The application has its own MySQL database with an assortment of tables that correspond to all the different details that must



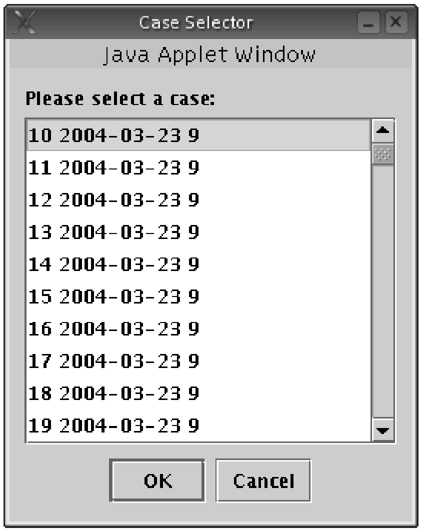
First Name: Bob
Last Name: Jones
Street Address: 3456 Washington Ave.
City: Topeka
State: KANSAS - KS
Zip Code: 66610
Sex: Male
Birthdate: 12/28/1960

Figure 5: This is the dialog box used to enter the information about a new subject.



Please select a person:
1 Jared Kite University of Nebraska - Lincoln Lincoln NE 0 m 1979-04-10

Figure 6: The dialog box used to select a subject whose information is already in the database.



Please select a case:

10 2004-03-23 9
11 2004-03-23 9
12 2004-03-23 9
13 2004-03-23 9
14 2004-03-23 9
15 2004-03-23 9
16 2004-03-23 9
17 2004-03-23 9
18 2004-03-23 9
19 2004-03-23 9

Figure 7: The dialog box used to choose a case associated with a subject whose data is already in the database.

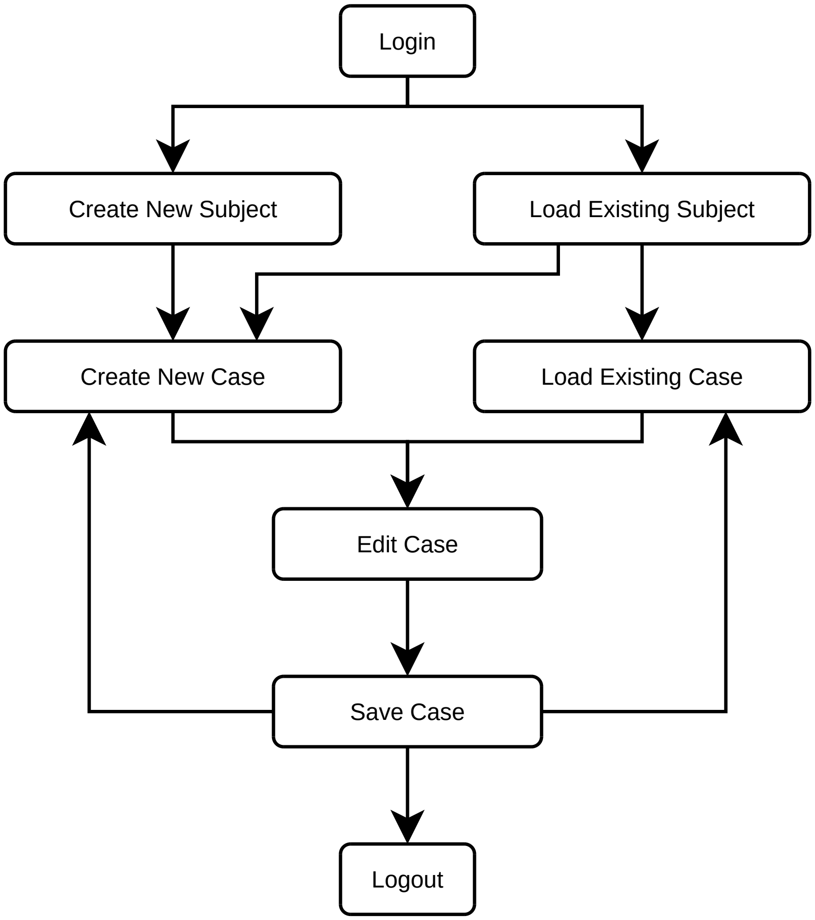


Figure 8: Shows the flow of the application from Login to Logout.

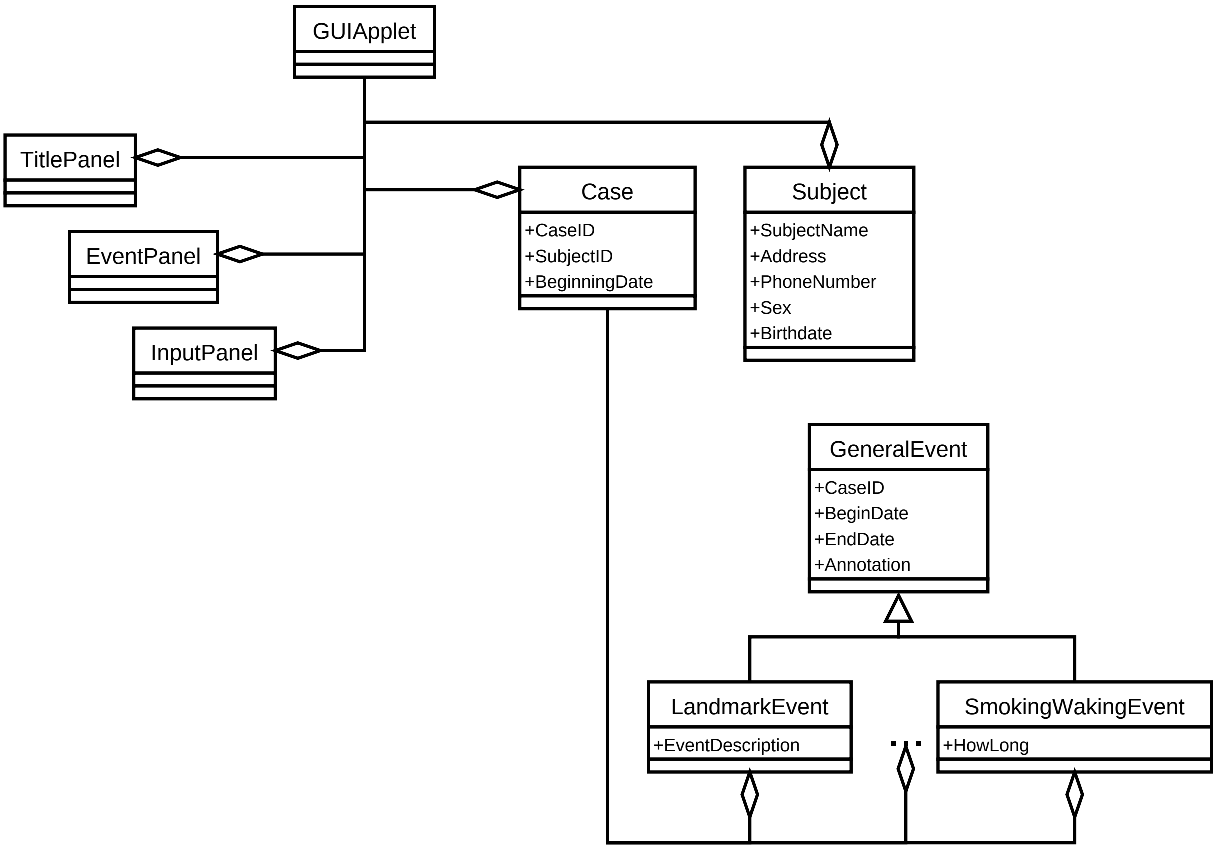


Figure 9: A detailed UML diagram of the classes used in the application showing their relationships and members.

be stored for a subject and zero or more corresponding cases. Figure 10 gives the details of each table in the database and the relationships between them.

Each subject is assigned an identification number when their personal data is entered. Every time the survey is administered to that subject, their identification number is associated with the case number. This association is found in the “cases” table. Each event that is filled in by the subject is associated with the case identification number and entered into the appropriate table.

With each event, we store a beginning and ending offset that corresponds to the number of thirds of the month that have passed since the beginning of the six month period. We also store the annotation and any information specific to the category along with the optional set of the beginning and ending days of the month. From the survey administration date and the offset we can reconstruct the actual time span of the event. The resolution of this is in thirds of a month unless the user entered the beginning and ending days for the events.

3.4 Intelligent Component

The third major component of our survey system, the Survey Assistant, is empty at this stage. As you can see in Figure 2, the Assistant has read access to the database and can communicate fully with the Applet GUI component. This will allow the Assistant to use the collective knowledge of the system to enhance interviews.

The usefulness of this component is to be explored in future research. We see much research potential for the Survey Assistant in the areas of Information Retrieval, Machine Learning, Data Mining, Human Computer Interaction and many other areas.

4 Conclusions and Future Work

We have created a computerized version of an Event History Calendar to study tobacco habits. Our application is state of the art in Event History Calendars, being specified by an expert. The framework is in place, allowing us to focus on ways to improve its capability to assist the interviewer.

There are three major advantages that our framework offers. Because our application is Web-based in nature, we can address a very wide audience and possibly move to self-administered surveys in the future. Second, we have the Computer Science skills and the interest in artificial intelligence to proceed with implementing the Survey Assistant. This has a lot of potential to improve data quality and information gain. Third, when the Survey Assistant is implemented, we will have reduced the need for expensive expert interviewers.

We see four progressive milestones in the development of this research: GUI with data entry, GUI with automated question generation, fully automated GUI for an adaptive interview, and an expert system assistant that will ask follow-up questions. Step one has been completed. Each new step adds to the capability of the system to compensate for a non-expert interviewer. In the final step, the interview can be completely self-administered.

Our immediate future includes research in step two, assistive interview. In this step we assume that the interviewer is a non-expert with some training. At this point, the application will propose useful questions for the interviewer to ask and allow the questions to be rated. We will collect a set of questions that are known to be useful in many of the conceivable interview situations and

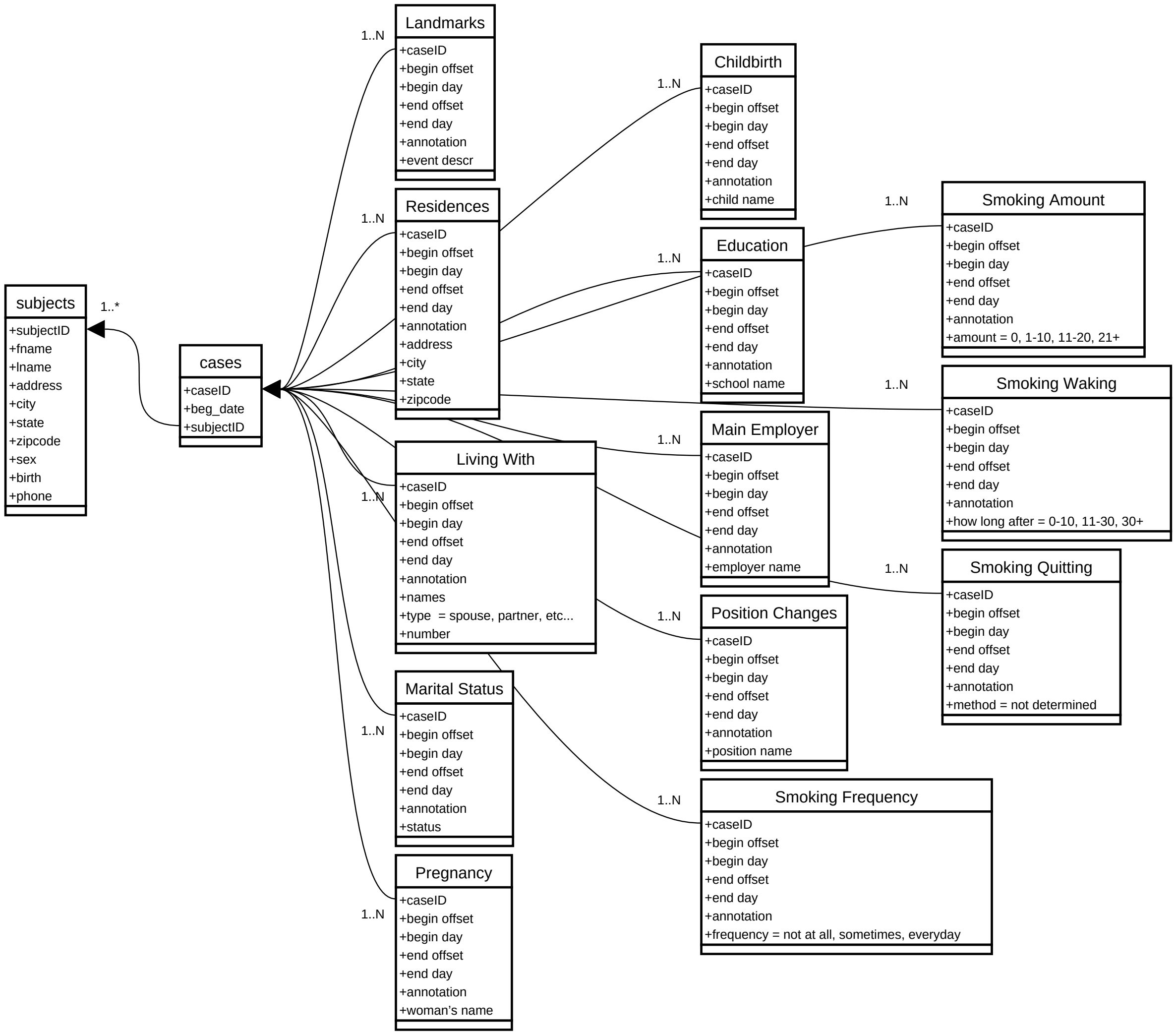


Figure 10: This is the organization of the database that stores all of the subjects and surveys for our survey system. Each titled rectangle here represents a single table in the database. It is obvious here that the organization of the database hinges on the “cases” table which relates each subject to zero or more instances of completed surveys.

use these as our initial set. New questions will then be proposed to the interviewer based upon the current information in the database.

The newly generated questions, we propose, will be generated using data mining including some machine learning. We will use data mining to recognize patterns in the data that are known and, hopefully, some that are novel. Questions generated to inquire about these patterns will be rated by the interviewer on their usefulness and they will be re-rated using relevance feedback.

An information retrieval technique for proposing questions will also be used. Using case-based reasoning, cases similar to the current one will be retrieved. The useful questions asked in those retrieved cases will be suggested to be asked again in the current case. These too will be subjected to the relevance feedback mechanism. This technique takes advantage of the work that was previously done in the set of retrieved cases and increases the number of potentially relevant questions.

There is much opportunity for research in this second step and the following. Following the second step, some of the research may include techniques from Human Computer Interaction, Natural Language Processing and Expert Systems. The advantage to this is that the survey system becomes more autonomous and the skill and training of the interviewer becomes less important. This means that the survey process becomes cheaper and many more people may be questioned within the same financial constraints. The final form of this is self-administered online surveys. Compared to the current costs associated with a survey, this is almost a no-cost solution.

5 Acknowledgments

We would like to thank the Gallup Research Center at the University of Nebraska, Lincoln for the generous financial support they contributed to make this project possible. We would also like to thank Dr. Bob Belli for contributing his survey design expertise and Xuan Lin for designing the GUI.

A Documentation

The Javadoc documentation for the source code of our project is available on the web at
http://cse.unl.edu/∼kite/gallup/javadocs/.

References

[1] Robert F. Belli. The structure of autobiographical memory and the event history calendar: Potential improvements in the quality of retrospective reports in surveys. *Memory*, 6:383–406, 1998.

[2] Robert F. Belli. Computerized event history calendar methods: Facilitating autobiographical recall. In *American Statistical Association Proceedings of the Section on Survey Research Methods*, Alexandria, VA, 2000. American Statistical Association.

[3] Robert F. Belli, William L. Shay, and Frank P. Stafford. Event history calendars and question list surveys: A direct comparison of interviewing methods. *Public Opinion Quarterly*, 65:45–74, 2001.

[4] Floyd J. Fowler and Thomas W. Mangione. *Stardardized Survey Interviewing*. Sage, Newbury Park, 1990.

[5] Deborah Freedman, Arland Thornton, Donald Camburn, Duane Alwin, and Linda Young-DeMarco. The life history calendar: A technique for collecting retrospective data. *Sociological Methodology*, 18:37–68, 1988.

[6] Daniela Giorgetti and Fabrizio Sebastiani. Automating survey coding by multiclass text categorization techniques. *J. Am. Soc. Inf. Sci. Technol.*, 54(14):1269–1277, 2003.

[7] Robert L. Kahn and Charles F. Cannell. *The Dynamics of Interviewing*. Wiley, New York, 1957.

[8] E. F. Loftus and W. Marburger. Since the eruption of mt. st. helens, has anyone beaten you up? improving the accuracy of retrospective reports with landmark events. *Memory and Cognition*, 11:114–120, 1983.

[9] Carsten Stig Poulsen. Artificial intelligence applied to computer aided interviewing. In *Proceedings of the ESOMAR/EMAC/AFM Symposium on Information Based Decision Making in Marketing*, Paris, France, 1993.

[10] Elizabeth F. Wiebe and Karl R. Landis. Evaluation of an electronic event history calendar. In *Proceedings of the Survey Research Methods Section, Americal Statistical Association*, Alexandria, VA, 2000. American Statistical Association.