
Design Project 1: Hello TiVo!

Hello. You've reached a real person. Say or press 'one' if you would prefer to speak with a computer.

—from a sci-fi movie of the future

In anything at all, perfection is finally attained not when there is no longer anything to add, but when there is no longer anything to take away.

—Antoine de Saint Exupéry

GOALS The first part of this exercise (10% of your Project 1 grade) is to become familiar with the rudiments of *PowerPoint XP* (for Windows or Macintosh). In particular, techniques will be practiced that will facilitate the production of design mockups. Besides *PowerPoint*, you will learn to use *SiteScape Forum* (SSF).

The second part of this exercise (90% of your Project 1 grade) is a team-developed design. Primary criteria for judging your design will be your degree of professionalism, your individual idea-stage designs, your ability to work collaboratively with your team, the quality of your presentation, and any other criteria that may emerge during interactions with “management.”

You will do this exercise in assigned teams (see the assignment list distributed during class and posted in SSF). However, Part 1 should be done independently by each person in the group. Part 2 should be done as a team project with a single submission for the entire team.

PART 1 *PowerPoint XP* is a part of the Microsoft Office Suite, and it is available to every IU student at a minimal cost through the computer store in the IU Memorial Union (you may download it for free as well). To learn the tool, check out from the library one of the many references on *PowerPoint*.

You are to learn *SSF* by using the Practice Forum and the Quick Start Guide. See: ssf.indiana.edu

The HCID 2004 forum is accessed at ssf.indiana.edu/msiegel



NOTE: If you have difficulty entering SSF, send an email note to me at msiegel@indiana.edu. You should verify this today. Your login name is your Network ID; your password is "hci".

Your Task

1. Enter the SSF HCID 2004 forum using the directions outlined in the *Course Overview for HCID 2004* handout. Then enter the "General Forum" discussion, write a reply to the first topic giving a brief description of you. **Label your reply with your name**, e.g., "Brian Horvitz". If possible, add a digital image to your profile (where you changed your password).
2. Open a *PowerPoint* file (a blank presentation with blank slides, that is, no pre-defined layout). Create a sequence of *PowerPoint* slides that begins with a title page: Three Parts of a Robot. Draw a solid box or border around the text. Each of these slides should have a plain white background. [While you're working on this file, save it to your online storage (CFS). Eventually, you will copy it to a reply in SSF for submission, in the "Project #1" forum, "Submissions-Part I" folder.]
3. The next slide should show the first part (that is, the robot, an arrow pointing to the part, and a label). The image of the robot can be found in the "Project #1" forum in SSF, topic "Robot image." You should upload it into your file.
4. The next two slides should show the second part labeled (note: the first part label is erased). For the second part, divide it into two subparts, for example, the left foot and the right foot. Show the left foot label. In the next slide, *add* the label for the right foot; that is, both feet are labeled now.
5. The next slide should show the third part labeled (the second part labels are erased).
6. Learn how to insert a menu after the title page. The parts of the menu should link to each of the three parts of the slide show.
7. Learn how to annotate each slide with a notes page. Write a simple notes page for each slide.
8. Do NOT turn this part into an elaborate art project. The purpose is to learn to use the features of *PowerPoint* that impact the design of mockups. Therefore, learn to draw boxes, circles, text, arrows, lines, etc., as well as how to systematically add and erase features. A completed example (without the notes) is seen in SSF.
9. The purpose of this exercise is to teach you how to add and remove parts to a display. You'll need to do this for your design mockups.



What to Submit for Part 1

1. A robot sequence as described above, with a menu and notes pages.
2. In the “Project #1” forum, upload your file to the “Submissions-Part I” folder. Title your submission with your name, e.g., “Brian Horvitz”.
3. In addition to uploading your file, print out your PowerPoint slides (don’t waste money on color prints!), with notes; one slide per page. Give this to your team facilitator for paper submission as well.
4. You may use your imagination on color and layout, but keep it clean and simple (also use **Verdana** as your font).
5. Be sure to write your SSF note reply describing yourself. Your note should be posted by September 5th. The *PowerPoint* file may be uploaded at any time before the due date (consult the timetable). Add a digital image of yourself to your SSF profile (where you changed your password).

PART 2 The second part of this exercise (90% of your Project 1 grade, excluding the collaborative score) is to create an interactive design that responds to the specifications provided below. Good design is user-centered. Think about the user’s model vs. the designer’s model.

Your team’s live presentation will be judged as well.

Your focus should be on user-centered interactive design. To emphasize this view, you will conduct a usability test on your initial design and then revise the design according to the test results. These revisions will reflect the needs of the end user, not your personal preferences.

You will work as a member of a team. See the team assignments posted in SSF.

Summary

Project 1’s goal is to design and usability-test a prototype for an automated voice activated TiVo guide by way of a customer’s phone. The purpose is to select a program (or programs) for future recording. See the TiVo site at www.tivo.com. This assignment also asks your team to design a protocol for visually presenting your automated dialog designs.



Overview

Your team is placed in the quickly emerging industry of voice application design. You work for a firm, InteractiveVOX (IV) that has negotiated a potential deal with TiVo, a new way to view and record TV. The TiVo web site has a full description of the product and service including a well-designed demo. In the “setup and support” section of the site you can find detailed descriptions of the product and services. As TiVo is considering the move to allow customers to select viewing by phone, you are charged with the task of developing a compelling prototype. This prototype will be used to convince TiVo to choose IV and your team to build their new voice site.

Imagine this scenario (there are others): You’re sitting in a restaurant with a friend. She mentions an upcoming program on TV. You realize you want to see the show but you won’t be home until after it’s over. Fortunately you own TiVo. By calling a special number from your cell phone (e.g., 800-555-TIVO) you can access TiVo’s new voice activated service. This will guide you through the process of selecting your targeted program for recording and future viewing. (Note: 800-555-TIVO is not real; it’s here for illustrative purposes only.)

Read through this entire document before beginning. Once you have developed an overall understanding of the assignment, you will undoubtedly have questions. Questions are integral to the design process. Try to get your questions answered as quickly as they arise, and remember that answers to many questions will require a creative response on your team’s part. As you gain a firm understanding of what the project entails, manage your time to accomplish the following seven objectives:

1. Narrow the project to its core components. Establish a set of assumptions and constraints, a design message for this project, and a list of feature specifications. (What’s a design message? We’ll talk about this in class, but it’s a succinct one sentence statement about the goal(s) of your project. It’s what you are trying to accomplish. It can be a phrase. It should be “owned” by every person on the team.)
2. One member from your team should submit to the online forum a set of design questions that will clarify the requirements.
3. Create several initial dialog designs per team member and choose one (or a combination of ideas) for the team to elaborate upon.
4. Create several methods for representing your dialog designs and choose one (or some combination) to evolve as the standard for communicating ideas within your team. Your team’s “dialog-flow representation protocol” will become the basis for your usability tests and for how your team will present its design to management.



5. Conduct a usability test of your team's design with at least three people. Based on your usability test results, revise your design as needed.
6. Make recommendations to management on your design, providing a rationale for major design decisions.
7. Create and deliver a compelling presentation – how the design is communicated, both on paper and in presentation, is as important as the design itself.

InteractiveVOX is a company specializing in voice technologies and voice interaction design. It does not specialize in interactive television. Your team is expected to become the experts on TiVo interactions - current and potential. Having said that, management may provide some general guidelines to you. Questions about issues in voice application planning, technologies, and implementation should be directed to the online forum. Questions about TiVo interactions may be asked there as well. Management is looking for dialog designs where a simple intuitive understanding of TiVo program setup is sufficient. A bad understanding of what customers might expect from a phone activated program setup service may be compensated by clever design concepts.

Important Note:

Attention should be given to the design *process* (including team dynamics) as well as to the design *product*. If the process is problematic, the resulting work will suffer. Each team member should think and behave on behalf of the team. If all team members do not share the same standards of excellence, try to resolve what to expect from each other early on. If work loads seem unfairly distributed, seek advice from your Mentor (an IV manager) and from Professor Siegel (IV COO).

Technology Summary

Speech recognition and text-to-speech synthesis technologies have evolved to the point where automated voice interaction is not only possible but is becoming common. There are numerous examples. Listed below are three. Every member of your team should call (toll-free) and try them:

- TellMe 1-800-555-TELL
- Amtrak 1-800-USA-RAIL
- HeyAnita 1-888-44-ANITA

The two main standards that have emerged in voice dialog implementation are SALT and VoiceXML. These W3C specifications are motivated by differing goals. SALT (developed mainly by Microsoft) is a set of tags that may be embedded into HTML types of pages to speech-enable those pages. By installing the SALT plug-in to your Explorer browser, you may talk with SALT-enabled sites. See <http://www.saltforum.org>, the official SALT site



to learn more about SALT. VoiceXML (developed jointly by AT&T, IBM, Motorola and Lucent) is designed specifically for running voice dialogs at large call centers. Your project would most likely be implemented in VoiceXML. See the official VoiceXML site: <http://www.voicexml.org> for more information.

Computer speech technologies have not advanced to the point of allowing natural conversation or any kind of machine understanding of what has been said. However, automatic speech recognition (ASR) is accurate enough to work well when the ASR engine is provided a list of utterances that may have been spoken (“Dictation Mode” sometimes works fairly well when an ASR engine has been trained on the speaker’s voice.). The ASR engine compares the caller’s utterance with items on an “expectation list” and the item on the list receiving the highest probability score, if that probability exceeds a threshold, will be returned as the choice or command the caller made. In other words, for ASR to work, dialogs must be designed in ways that direct conversation so a list of possible caller utterances may be anticipated at each juncture. When a caller’s response is not recognized from a list of possible commands corresponding to pre-determined actions, the caller’s utterance is meaningless and only a default, “utterance-not-recognized” action may be taken.

This is the single most important aspect of voice interaction technology for a new dialog designer to understand. Let’s consider it further with a specific example. Say you are designing a system where you know you will want to direct a caller to the subroutines of Lisa, Sarah, or Daniel. You decide the utterance a caller speaks will go to a subroutine with the name of the person who owns the subroutine – i.e., a caller will say “Lisa” to go to Lisa’s subroutine or “Dan” or “Daniel” to go to Daniel’s subroutine and so on. You must plan a prompt that will influence the caller to reply with either “Lisa”, “Sarah”, or “Daniel”. A prompt such as: “Say one of the following: Lisa, Sarah, or Daniel” would be much more effective than a prompt such as: “What do you want?”

Voice technology has other attributes. When a caller responds, their response may be recorded as an audio file and that file may be passed to any program. For instance, at a music information retrieval site, a caller may hum the melody of a song they heard on the radio to have that site return the song’s artist and title along with options on how to purchase the CD (this site is under development by researchers affiliated with Sony).

Voice dialogs may be generated on the fly. That is, information collected from a caller in earlier dialogs or earlier visits may be used in creating new dialogs. For instance, if you determine a caller is a caller for whom you have information stored in your database, you may use that information in your prompts. “Oh, Mr. Jones, it’s good to hear from you again. Did you receive the book you ordered through our site?”



Voice applications may pass a caller to another phone number. For example, in a voice activated restaurant guide, you might offer the caller the option of connecting directly to a restaurant once they have decided where to make reservations or have decided on a place for where they need driving instructions.

Voice applications may initiate the call. You may program your “automated digital assistant” to call you in the mornings and read you your day’s schedule and e-mail. Or you may program your assistant to call everyone in your phonebook to remind them of your birthday!

Voice applications know the number the caller is connecting from. You know a caller’s area code and can work to tailor your dialogs to location-specific information based on that area code.

There are speech recognition technologies geared at recognizing the gender of a caller, the general age of a caller, and even the emotional state of the caller. These technologies are probably not yet robust for widespread use, but may be considered as your team develops its dialog representation protocol.

This is by no means a complete list of everything that can be done with today’s voice application technologies. However, it should help to give an idea of the possibilities. Your team is not expected to exploit all or even most of these options as it designs its TiVo guide; but keep them in mind as you develop your representation protocol.

Design Challenges

You must address 3 main issues with your design:

1. Callers may search for and specify any program that would record on their home TiVo. (Note: the caller may call from anywhere.)
2. Callers may learn about what’s possible for viewing, but there are practical limits.
3. Provide an overall concept and/or additional ideas that will convince TiVo to choose your design over the other teams’ designs

How callers ask for information and how it is provided is up to your team. The way in which you structure your information is important, and this may differ from how it is done at home with a television screen. Also keep in mind the motivation the primary motivation for your design. What is it?

There are a number of issues with which your team does not need to be concerned:



- You do not need to consider existing TiVo technical architecture. Assume that TiVo engineers will correctly program the TiVo from the output of your IV interaction.
- You do not need to become an expert in speech processing technologies. Everything you need to know of the technical aspects of voice application design was spelled out earlier in this document or will be addressed in the online forum.
- You do not need to implement your design. The flow of dialog must be carefully planned out before being coded. It is this planned dialog which is to be turned in. Note: Your dialog must be complete and precise.

Voice dialog design is a relatively new field with few well-established practices and methodologies. Your team is expected to forge ahead and develop a dialog-flow representation protocol in addition to your actual design. This protocol will be used to visually communicate your ideas to each other and to TiVo. Your team's protocol should strive to be self-explanatory, not requiring a long complicated legend, and it should be free of logical errors leading to potential ambiguities in control-flow. Some kind of flow chart is generally expected, where the computer's question or prompt leads to a caller's response and the caller's response in turn dictates the computer's next prompt or action. See Figure 1 for a very broad reference. Colors, shapes, embedded actions, global actions, and the like may be used. Imagine your protocol as forming the basis for a future graphical development environment. Devising this protocol is a difficult design project in its own right; your team should simply try to come up with some improvements (or a completely new system if you think of something better) over Figure 1. (See the Project 1 folder in SSF for this design sketch.)

Some Voice Design Heuristics

Here are some things to keep in mind as your team designs its dialog:

- A caller must maintain the entire navigation in their imagination. Memory and cognitive demands should be kept reasonable. Try not to ask the caller to remember things where they will need to jot notes on paper.
- The caller should always have a clear understanding of where they are in the navigation. A caller should be able to start over or go to a familiar place relatively easily.
- Provide instant feedback. When a caller utters something, immediately let them know what the computer understood the utterance to be. If



the computer misinterpreted the utterance, it's better to correct it quickly than let the caller mistakenly think they are in the right subroutine.

- Try to keep caller options distinct sounding. Automatic speech recognition engines are more likely to make mistakes when different options sound similar. For instance, it's hard for a speech engine to distinguish the phrase "it's easy to recognize speech" from "it's easy to wreck a nice beach."
- Spoken interaction is bottle-necked and slow. Because of this, don't seek to provide more information than is called for. People can use other means (e.g., the web) to do more extensive research.

Usability Testing

Usability testing is an invaluable tool for interaction designers. Often the design process will overlook or even mask potential problems that testing can help to discover. Though usability testing will probably not be able to fix a bad design, it will tell you if you have a bad design and it can be very helpful in fine-tuning a good design.

Once your initial design is mapped out in your dialog-flow protocol, you can use a member of your team as the computer to interact with a caller. Recruit 3 people to participate in testing your team's dialogs. Before beginning, tell them about your assignment and tell them to imagine a scenario where they wish to program their TiVo (like a VCR but better) to record their specified selection. Then read the prompts and respond to replies exactly as a computer would respond following your flow chart. Try not to break out of this test mode until the entire call is complete. If you have to stop in the middle of the test to explain something, your design is likely broken. It is preferable to do your tests over an actual phone so that visual cues do not influence communication. If you can audio record your usability phone calls, you will be at a strong advantage because it is much easier to focus on reading the prompts correctly while not having to think about taking notes or remembering what went wrong. Otherwise, two people should conduct the test—one to act as the computer, the other one to record problems and caller behavior.

When you talk to the subject, have your team listen in on another line, or tape record your interaction for later analysis. Alternatively, you may do a "live" usability test, where you interact with your caller in person in front of your team. **YOUR ENTIRE TEAM MUST ANALYZE THE USABILITY TESTS TOGETHER.**

You may submit audio recordings of these interactions (.wav format) in lieu of a write-up of your usability testing procedures. You may also use these



recordings in your presentation to management. Of course, you'll still need to write up your results.

What to Submit for Part 2

This project is intended to be fun but it is difficult also. Creating a complete design is likely to take more time than is allocated. Completion is not the only goal or even the most important goal. How your team tackles the problem and solves fundamental aspects of the problem is more important than a weak solution addressing many issues. Quality, not quantity is demanded here. To put this more concretely: it is possible to earn an A+ on this project by creating an excellent solution to the core problems.

Here is what is to be submitted:

1. Design statement: the design message (one sentence maximum) along with assumptions, constraints, and feature specifications – should be less than a page long (as short as possible while saying everything needed).
2. Design argument: explain why your TiVo information / dialog design is the best possible – should be about a page long (as short as possible while saying everything needed).
3. Dialog design protocol: A description of your team's dialog-flow representation scheme (the scheme may be several pages long). Hopefully the representation scheme or "protocol" your team designs will be self-explanatory to the point of not needing a description, but describe important features – should be well less than a page (as short as possible while saying everything needed).
4. A minimum of one "idea-stage" dialog design per team member. These are rough sketches at best. Please be sure to label who did what.
5. Initial prototype – your team's design before usability testing. This, along with the final prototype, should be presented in your team's dialog-flow representation protocol.
6. Usability testing: analysis of what was learned in usability testing – should be about a page (as short as possible while containing all important information). Audio recordings of your usability tests may be submitted in lieu of the written component (.wav files posted to your team folder in the on-line forum. Double-check that they uploaded okay and are not corrupt; you are responsible for ANY problems – to be safe, you may opt to put them on a CD and turn it in as well).
7. Your team's modified/improved prototype - the final prototype described in your team's design protocol.
8. A power-point presentation delivering your ideas to management – about 10 minutes (about a dozen slides).



9. Each team member's review of each of their teammates – in sealed envelopes.

Items 1, 2, 3, and 6 should be combined into a single word document (with the different sections clearly labeled) called “Hello TiVo Prototype,” or something similar. This document, along with a hard copy of your team's PowerPoint presentation, hard copies of all designs, along with each member's team reviews should be put into a single large envelope and turned in by the due time and date. Your design documents along with your PowerPoint presentation should also be posted to SSF the day the assignment is due and before the due time. Students also should save copies of the idea-stage designs they submit. The facilitator should back up all files for safe keeping.

The form for evaluating your teammates will be in the handouts section of SSF.

You will be graded based on your degree of professionalism, your individual idea-stage designs, your ability to work collaboratively with your team, the quality of your presentation, and any other criteria that may emerge during interactions with “management.”

