ADAPTIVE LEARNING ENVIRONMENTS: Formative Evaluation

Contents

1. Overview of Formative Evaluation
2. Case study: Standup
3. Methods
4. References

Also see lecture 6 on Formative Evaluation in Intermodeller

Some material based on Ainsworth's AIED 2003 tutorial on Evaluation Methods for Learning Environments, see AILE course web page and link:

http://www.psychology.nottingham.ac.uk/staff/sea/Evaluationtutorial.ppt

1. Overview of Evaluation

Stages of system evaluation…

1. Task and requirements analysis
2. Design
3. Evaluating design
4. Prototyping
5. Re-design and iterate
6. Internal evaluation of content
7. Satisfaction of design requirements
8. Usability
9. Effectiveness
10. Conclusions r.e. hypotheses tested

What is being evaluated?

The **design**?

The **usability of the interface**?

The **correctness** of the system knowledge?

The **accuracy** of the user model?

The **model of theory** implemented in the system?

The **performance** of an algorithm?

The **effectiveness** of the system?

Does the system do what we say it does?

Or is the system being used to evaluate **some aspect of educational theory?**

Goals of evaluation

To assess the extent and accessibility of system functionality:

Does it satisfy system requirements?

Does it facilitate task completion?

To assess user experience of the interaction:

Does it match user expectations?

How easy is it to learn?

How usable?

User satisfaction?

Does it overload the user?

To identify specific problems with the system:

Are there unexpected results?

Does the system cause confusion for users?

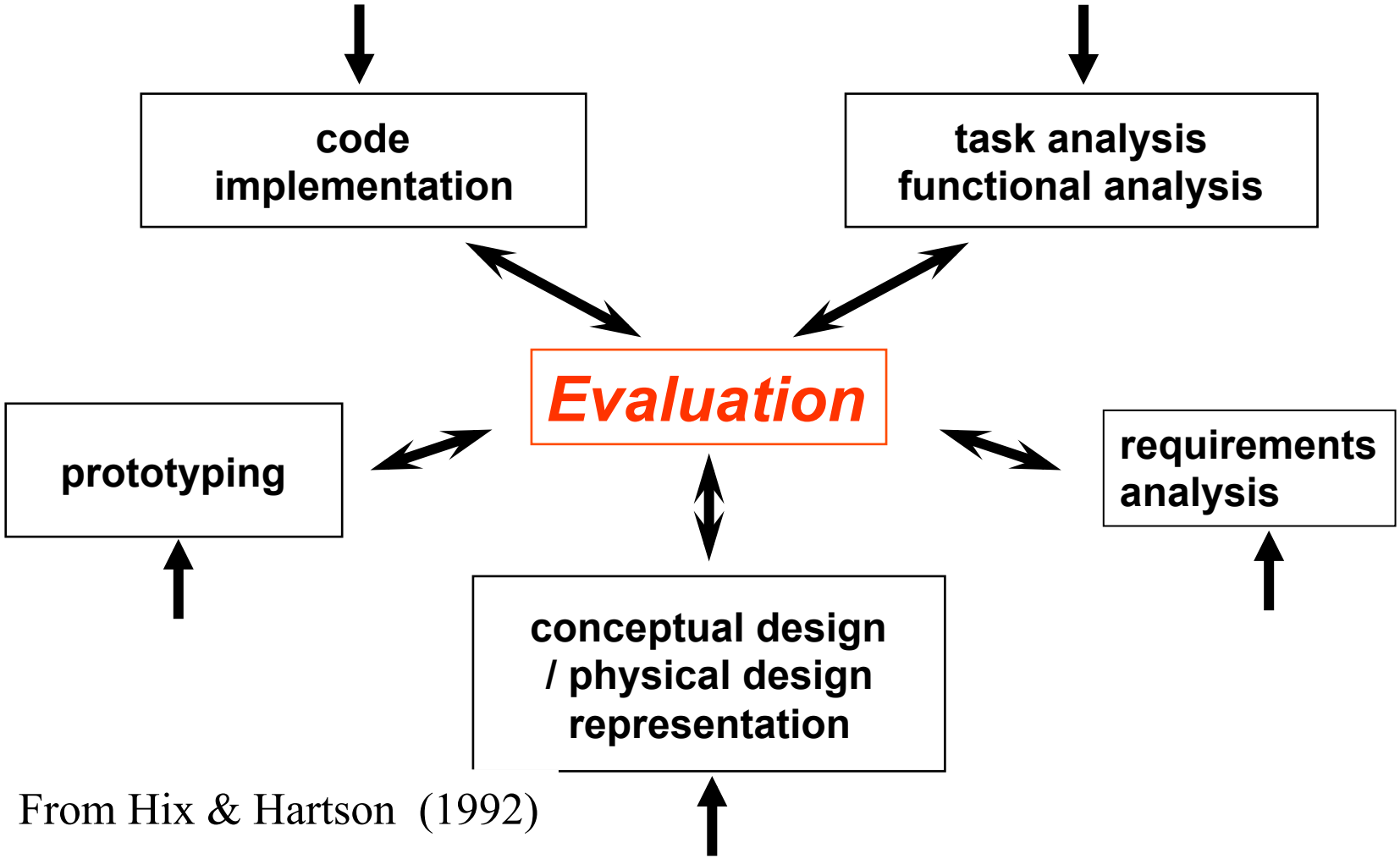
Other trouble spots?

Evaluation Points of View

1. Educational technologist/designers point of view
2. Teacher, Educational expert, Domain expert point of view
3. User, student point of view

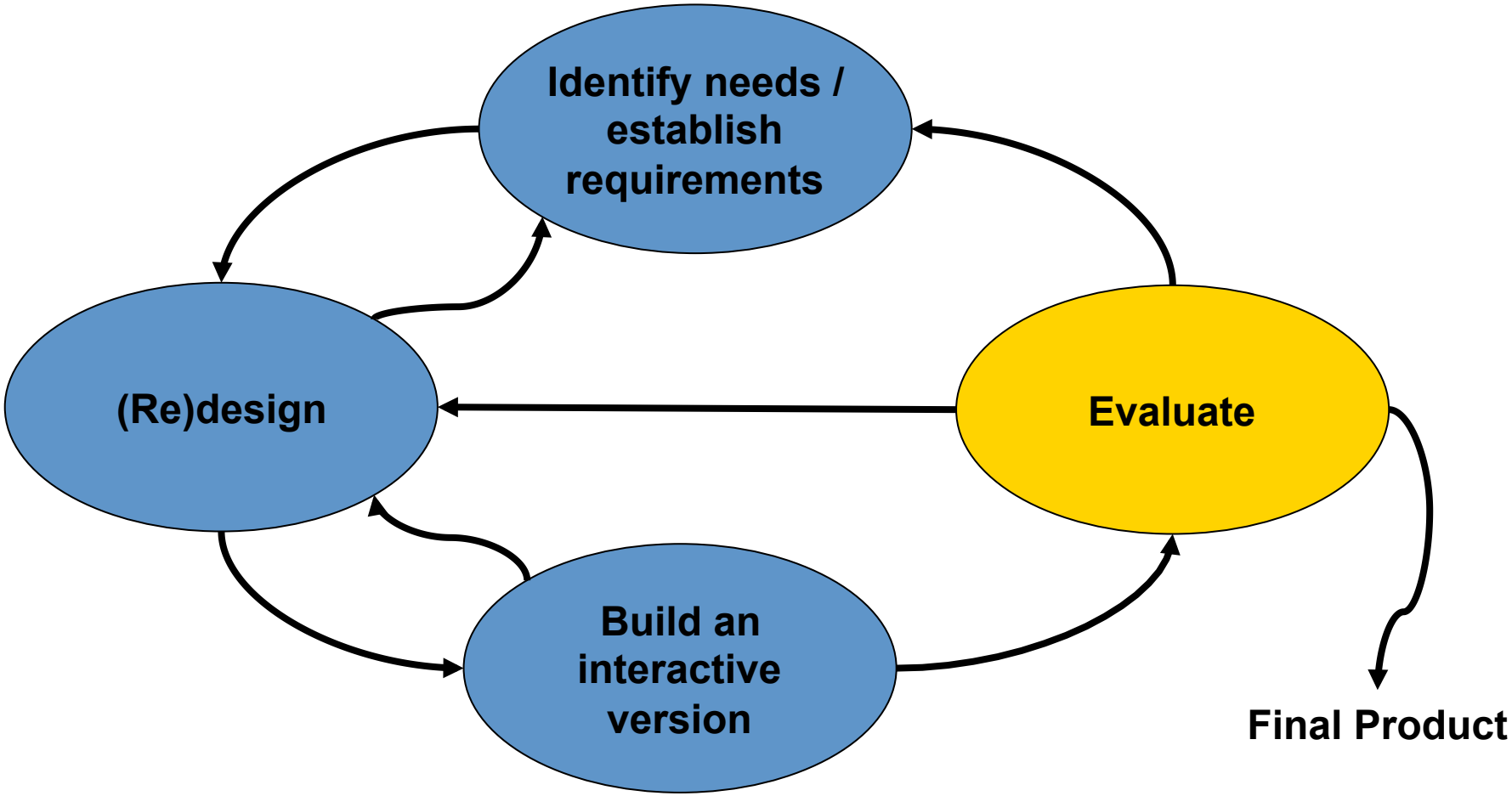
[these all have differing requirements and different measures of success.]

An iterative view of system development
(from Waller, 2004)



From Hix & Hartson (1992)

Waller (2004) summarises…



Common Evaluation Methods

Task analysis
Observation
Cognitive Walkthrough
Mock-ups
Protocol analysis
Wizard of Oz
Interview (structured/unstructured)
Questionnaire
Focus groups
Heuristic Evaluation
Expert evaluation
Sensitivity Analysis
Self Report
Post-hoc analysis
Logging use
Dialogue mark-up and analysis
Manipulation experiment
Sentient analysis

Methods – Class Activity

We went around the class, each nominated a method and defined it.

For each we briefly discussed it, and where it might be used.

If you missed this, then please look up the various methods

We will talk a bit more about pros and cons of various methods later in the lecture....

What sort of study?

Observational?

Survey?

Experiment?

Field study?

Participants?

Students? Teachers?

Technologists? Designers?

Domain experts? Pedagogical experts?

Formative v. Summative Evaluation

Formative Evaluation:

- throughout design and implementation
- incremental
- assessing impact of changes
- frequently qualitative

Summative Evaluation:

- on completion of each stage
- assessing effectiveness
- frequently quantitative

Qualitative v. Quantitative Data

Qualitative

Descriptive data

Based on system behaviour or user experience

Obtained from observation, questionnaires, interviews, protocol analysis, heuristic evaluation, cognitive and post task walkthrough

Subjective

Quantitative

Numerical data

Based on measures of variables relevant to performance or user experience

Obtained from empirical studies, e.g. experiments, also questionnaires, interviews

Amenable to statistical analysis

Objective

Analysis Methods

Qualitative v Quantitative

Statistical?

parametric v non-parametric

Data presentation methods?

graph, bar chart, pie chart, table,….

Common Measures (Dependent Variables) *(from Ainsworth, 2003)*

Learning gains

Post-test – Pre-test

Learning efficiency

i.e. does it reduce time spent learning

How the system is used in practice (and by whom)

ILEs can't help if learners don't use them!

What features are used

User's attitudes

Cost savings

Teachbacks

How well can learners now teach what they have learnt

(step back) Approach to Design

Informing Design of an ALE

In small groups reflecting task choice in assignment 2 (teach a sport, or English) list ways that the design could be informed:

- Who would be involved (stakeholders)?
- What resources/materials would you use?
- What methods?

Designing Technology:

We feel that it is important that the design of technology for learning and communication is:

- Evidence based – derived from theory and previous research
- Informed by empirical studies
- Designed in participation with stakeholders

Theoretical underpinnings: cognitive development, pedagogy, therapy

Technology design methods for learning by young children, with special needs

Importance of user-centred and participatory design

Early user involvement in the design of software systems is essential if the system is to be usable

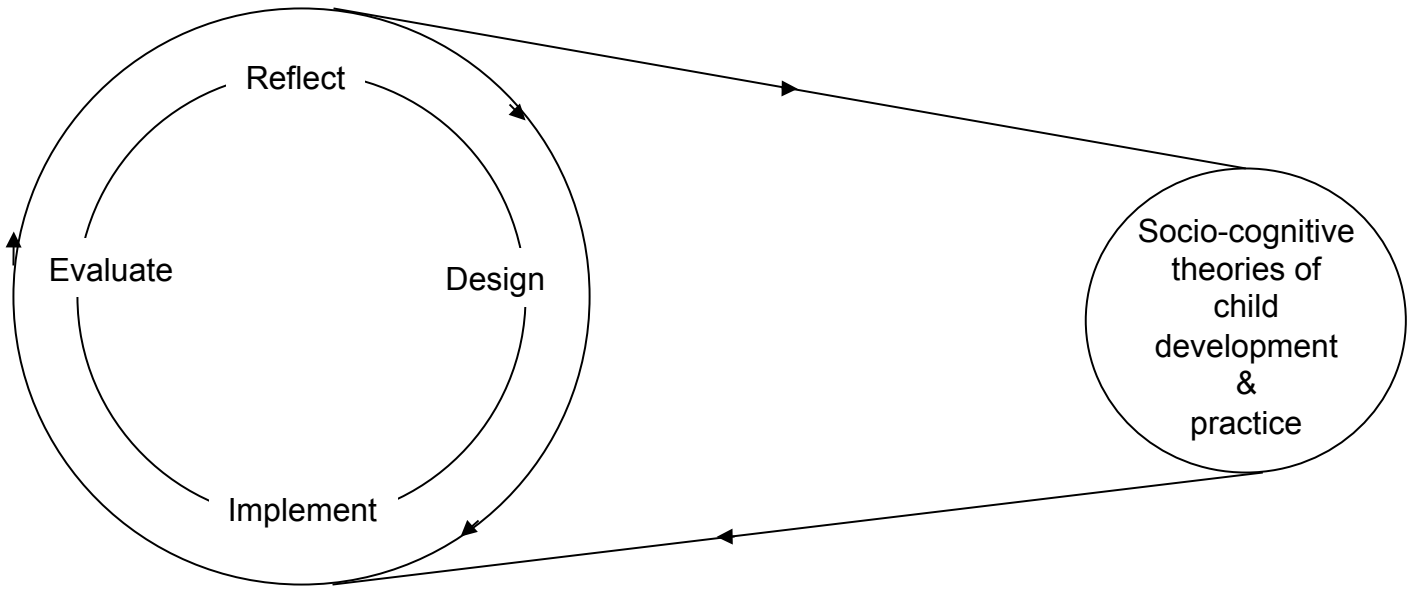
(Preece, et al, 1994; Shneiderman, 1998)

Moving from "system-centred" to "user-centred" design has enabled great improvements to be made in the effectiveness of user interfaces *(Wood, 1998)*

"The UCD approach is vital in the area of assistive technology ….. this approach presents a challenge when designing for people with severe communication impairments who may not yet have acquired effective communication strategies"

(Waller et al, 2005)

Persistent Collaboration research methodology



Participatory Design (PD)

Based on the simple but powerful idea that ***people who use particular tool or system should have a say in how it is designed and what it does.***

- Not about a pleasing look-and-feel to the end product (though PD may have this result)
- Not just a "user feedback survey" or a focus group
- PD begins when a project begins!

True goal: **incorporate *system stakeholders from the beginning***

- Incorporate existent practices, perspectives, and needs
- Stakeholders as active and valued partners, rather than passive recipients of a finished product
- Especially for learning technologies, may be a disconnect between what experts, users, and others think is most beneficial.

PD as an ethical issue: Design ***with, rather than design for***

Participatory design (PD) methods:

Helps ensure product designed meets users' needs and is usable;

Increases likelihood of suitability and sustained use;

Engaging teachers in design and evaluation:

- improves teacher's understanding of technology and research,
- enables them to contribute to design as more critical users.

Software design benefits from input from learners:

- **all children (incl. those with complex communication needs and severe physical disabilities) can take part in the PD process** (e.g. Frauenberger et al, 2013), and in evaluation of software (Waller et al, 2009), thus empowering users.

Designing for children with special needs is challenging and provides a test ground: software accessible to such children usually ensures accessibility for typically developing children.

2. Case study: formative evaluation of Standup

STANDUP

System
To
Augment
Non-
speakers'
Dialogue
Using
Puns

Need for language play opportunities

Word play is critical part of language development

- typically-developing (TD) children enjoy jokes and riddles
- provide opportunity to practise language, conversation and social interaction skills.

Jokes

- are a type of conversational narrative
- play an important role in the development of storytelling skills.

Role of punning riddles in language development

- pragmatics => turn taking, initiation etc.
- vocabulary acquisition

Children with speech and/or language disabilities do not always have language play opportunities.

Augmentative and Alternative Communication (AAC)

AAC: augmentative or alternative ways to communicate for people with limited or no speech.

e.g. people who experience cerebral palsy, multiple sclerosis, stroke

Most AAC devices based on the retrieval of pre-stored linguistic items, e.g. words, phrases and sentences.

Humour and AAC

- prestored rather than novel jokes
- order of retrieval and pragmatic use
- little opportunity for independent vocabulary acquisition and word play
- research mainly into enjoyment and fun

Little research on role of humour in AAC or the role it plays in developing language skills.

Standup goals

To build a tool that helps children with complex communication needs (CCN) to play with language:

1. **generate novel puns** using familiar vocabulary,
2. **experiment** with different forms of jokes.
3. provide **social interaction** possibilities
4. **go beyond** the "needs" and "wants" of **AAC**

Such a tool should be:

- **Interactive**: speed, efficiency
- **Customizable**: extensible
- **User-centred** design for CCN-specific interface
- **Appropriate** (e.g. not unknown vocabulary)

Could we develop a usable interface to a joke generator?

Initial Requirements

User Requirements

Group 1: Children with Complex Communication Needs (CCN)

(limited access due to fatigue and time constraints)

- Impaired language use
- Not impaired intelligence
- Literacy level below expected for age
- Possible physical impairment (e.g. cerebral palsy)

Group 2: Typically developing children (TD)

- No language impairment
- Expected literacy level

Experts:

- Teachers, parents, speech therapists, carers
- Plus **CCN Adults** as expert users

Usability Requirements

Not too many key presses

Easy to go back if make unintended selection

Different levels of access to manage language skills and possible progressions:

Vocabulary (*measured by word frequency)*

Task difficulty (keyboard input harder than simple selection)

Joke type (partial word matching harder then homophone substitution)

Accessible to all users by scanning, switch, touch screen or direct access

Assume use at home or school (with help to set up)

Speech access (generation, not recognition)

Joke Generation Tool: Functional Requirements

Be able to generate jokes:

1. Based on a **topic Food > Vegetables > Onion**

   What kind of vegetable can jump?

2. From **keyword**(s)

   Using **car** and **sandwich**

   What do you get when you cross cars and sandwich

3. From **templates bazaar:** How does a ___ ___?

   How does a whale cry?

4. From **Favourite Jokes** list

   How is a car like an elephant?

Joke Generation Tool: Functional Requirements

Be able to generate jokes:

1. Based on a **topic Food > Vegetables > Onion**

   What kind of vegetable can jump? A spring onion.

2. From **keyword**(s)Using **car** and **sandwich**

   What do you get when you cross cars and sandwiches?

   Traffic Jam

3. From **templates bazaar:** How does a ___ ___?

   How does a whale cry? Blubber blubber.

4. From **Favourite Jokes** list

   How is a car like an elephant? They both have trunks.

Technical Requirements

Templates, schema and lexicon to generate joke

Lexicon related to topic (by some method of classifying)

Appropriate for Young Children

- No Unsuitable Words

Lexical information on word frequency

Informing the Design of the Interface

Interface design: initial feedback

Speech and language therapists (SLT) in two **focus groups** discuss initial requirements and general design principles:

- **Interview**
- **Task analysis**
- **Paper mock-ups**

Feedback:

- assumed too high a level of literacy and too much reliance on text
- need picture language interface
- suggests various ways such a tool could be used
- were enthusiastic and wished to be involved further

Developing system requirements and alternative conceptual designs

1. **Difficult to use real target users (children with CCN):**
   - hard to communicate needs and opinions
   - would be easily fatigued

2. **Adults with similar difficulties**, but better technology and communication skills were used, **as expert end-users**

Composite interface of possible joke-generating sequence, using sequence of interface screens:

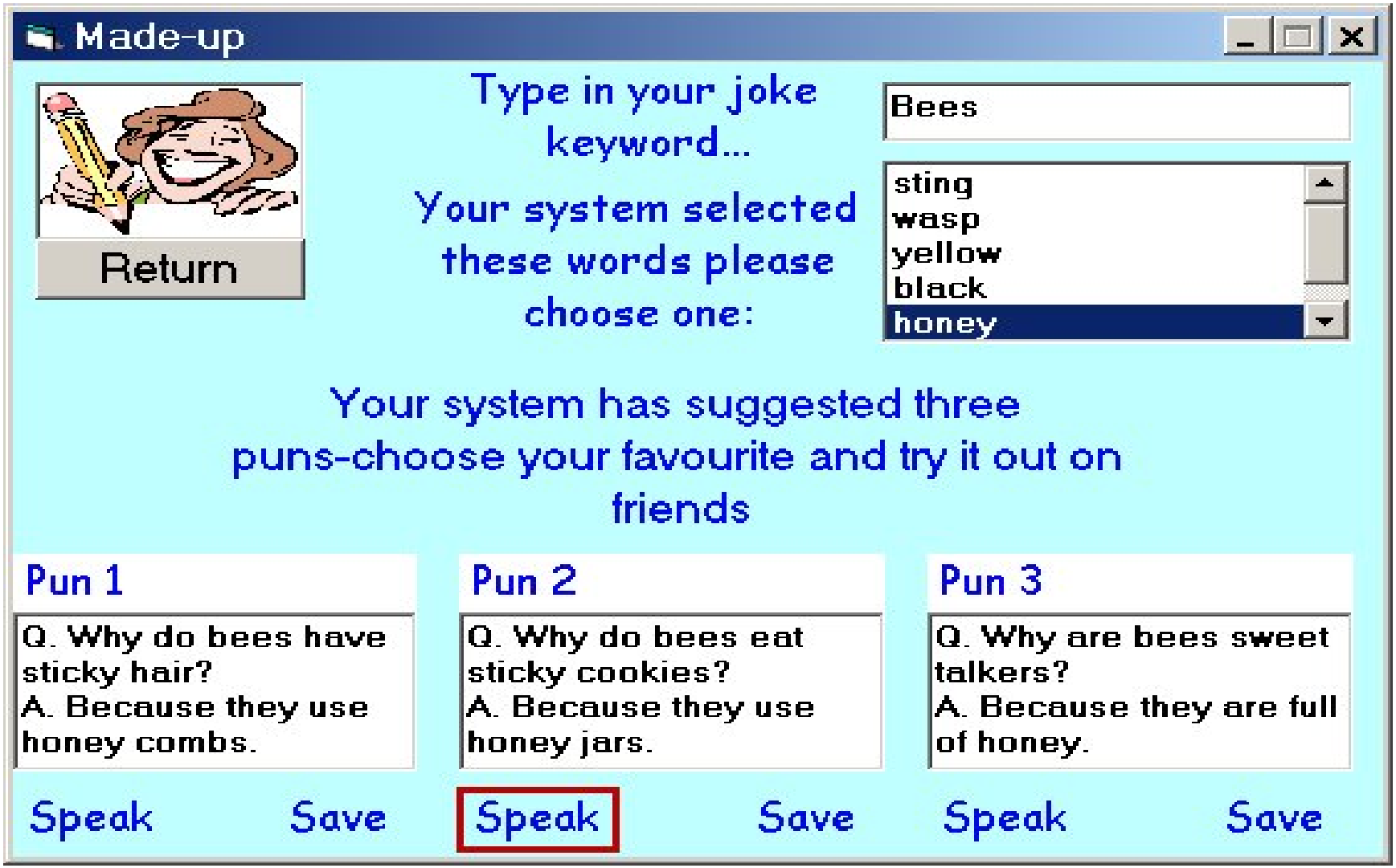
a. **"highly literate"** with text-based interface

b. **"highly pictorial"** based on journey metaphor

Two different system prototypes evaluated by:

- Five Speech and Language Therapists (SLTs)
- Two adults with CCN as end-user experts

‘Highly literate’ prototype



Data collected from SLTs on '*highly literate*' prototype

- It looks boring
- It is not how we teach early literacy skills
- It needs to be much more stimulating
- It needs to be able to give early rewards and this looks like it could be difficult
- I realise there will be auditory signals but it is still very unappealing for a child
- It doesn't appear to encourage use
- A small minority may be able to use something with this much language
- It looks fine for kids without any physical or learning difficulties

Revised User Requirements

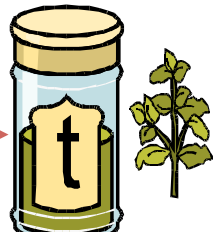
Vocabulary - Appropriate for Young Children

- No Unsuitable Words

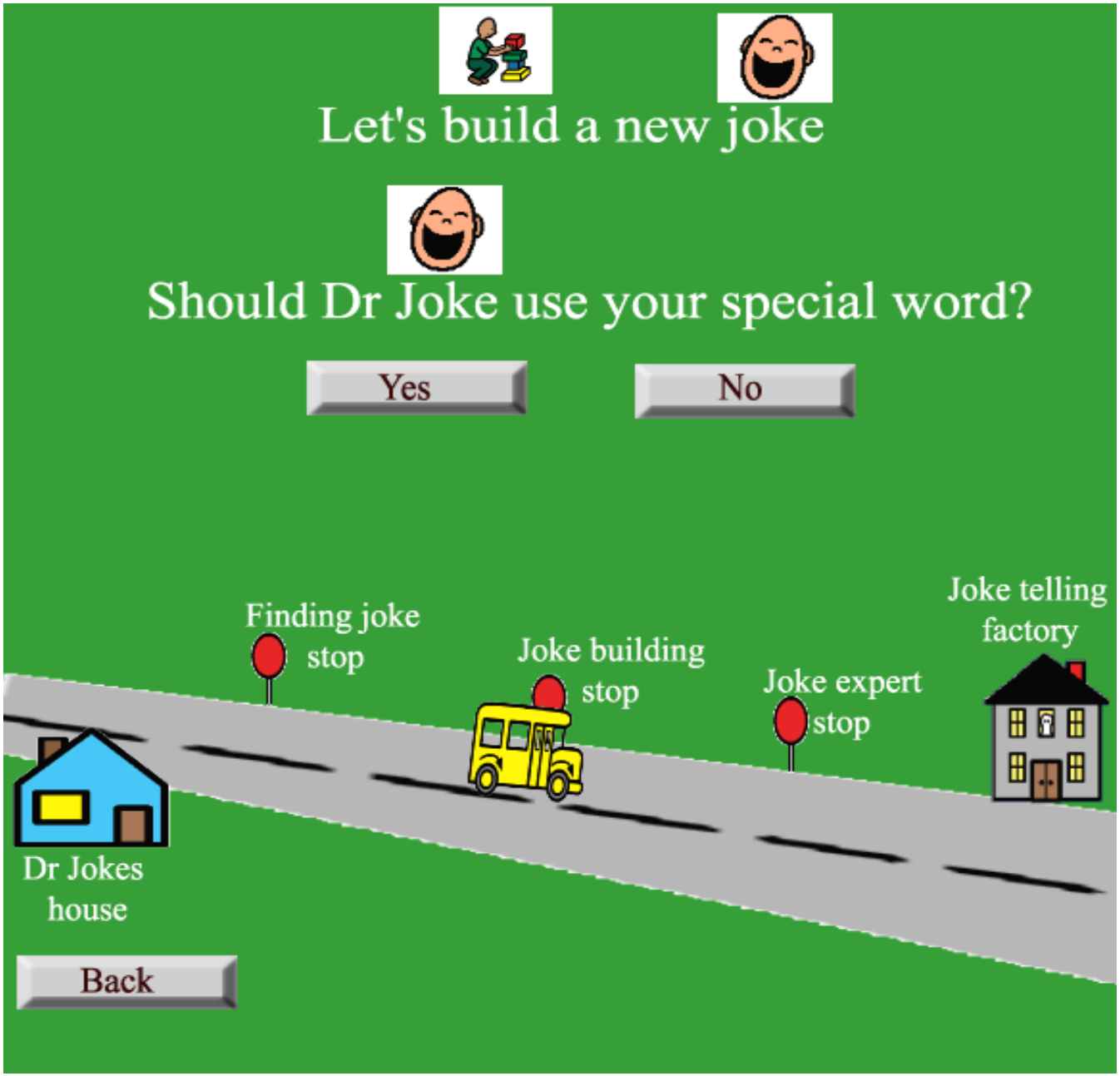
Appropriate for Children with Emerging Literacy

- Preference for Familiar Words
- Speech output
- Symbol support to support interface test and scaffold literacy using Rebus and PCS symbol libraries e.g.:

"*market*" → 

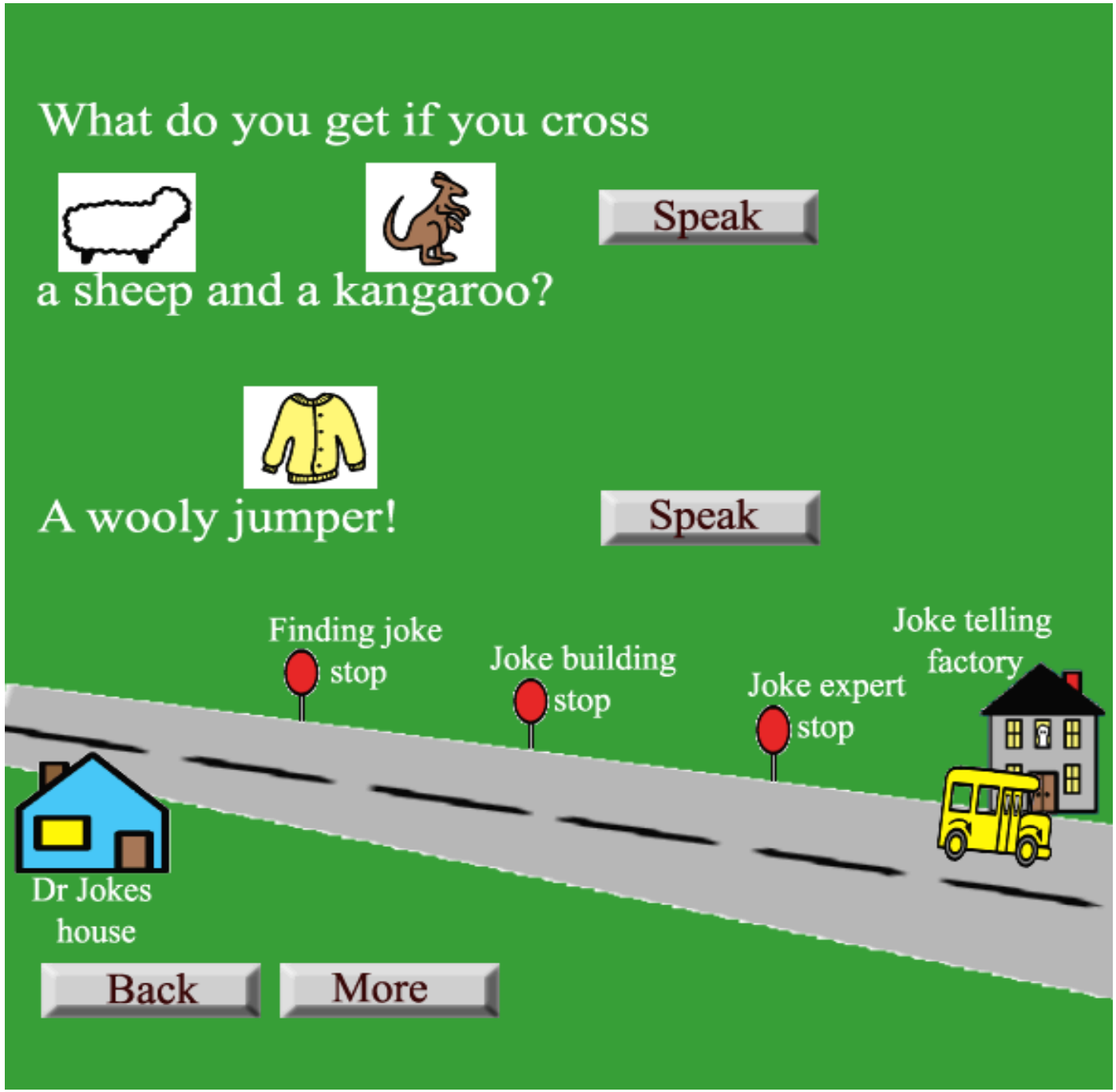
"*thyme*" → 

Access to jokes using subjects – lexicon grouped into subject-areas (topics) and clustered into a hierarchy



'Highly Pictorial' Prototype

Interim Home screen for journey metaphor



‘Highly Pictorial’ Prototype

Interim screen for journey metaphor showing joke and answer to be ‘spoken’ by speech synthesiser

Data from expert end-users

Design:

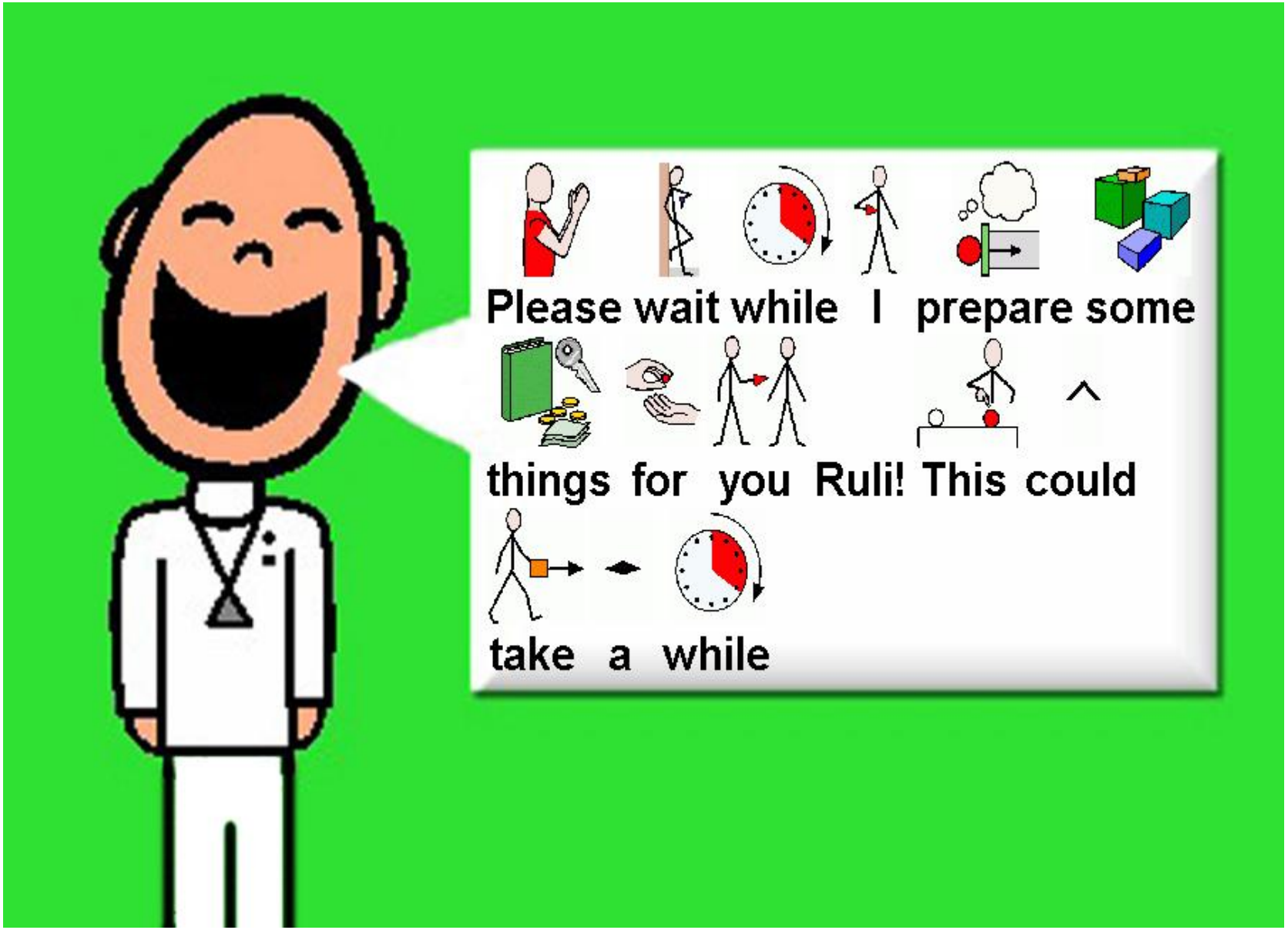
- **Videotaped, usability test-scenarios**
- **Semi-structured interview: closed questions** (questionnaire inappropriate,
- *Two short sessions to avoid fatigue*

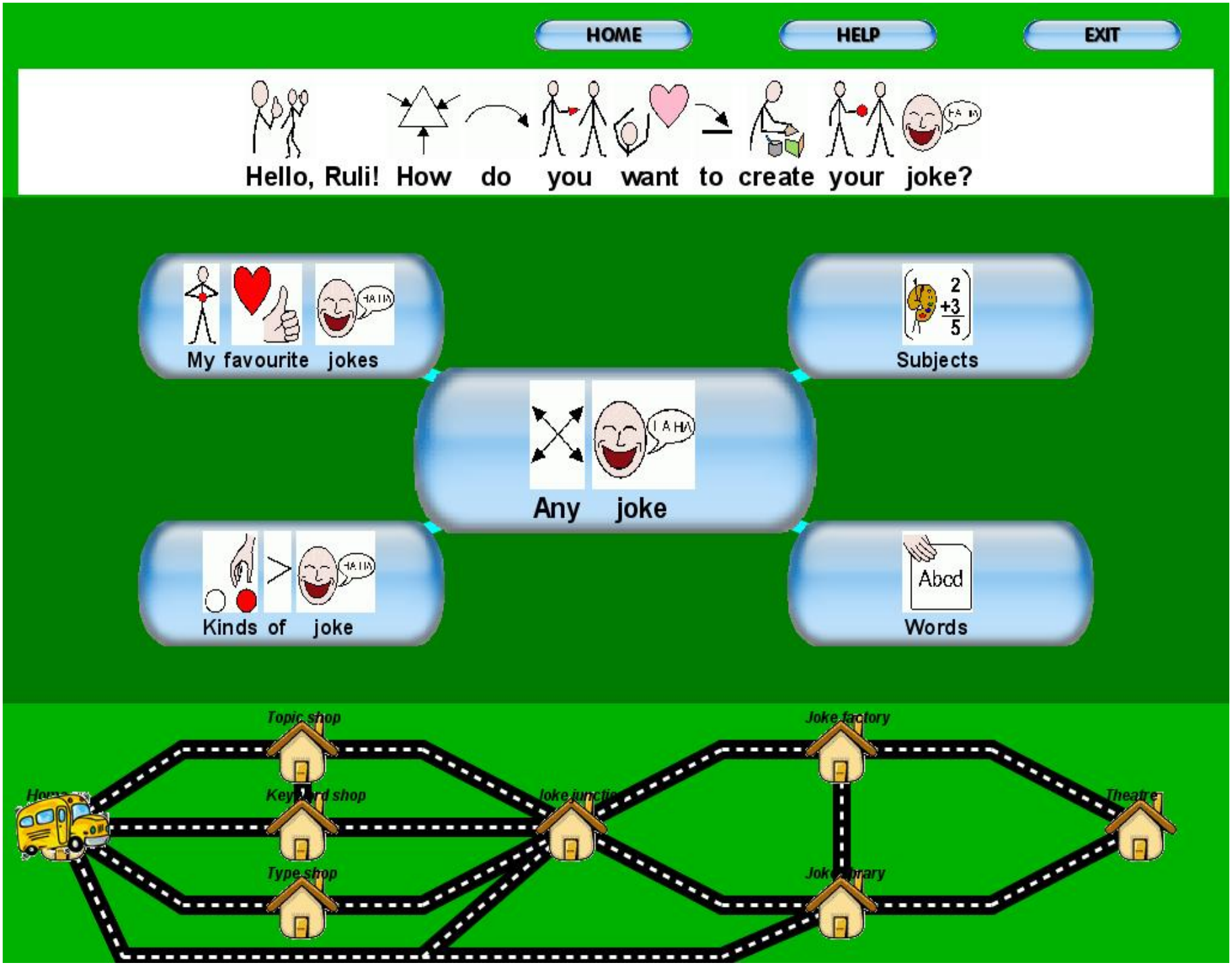
Usability issues:

- able to complete the set tasks with some ease
- able to retrace steps by pressing the "Back" button
- understood concept of telling the first part of joke then punchline

Design feedback:

1. Preferred pictorial journey interface to text-based one
2. PCS symbols useful for word reinforcement
3. But users should have option to switch PCS off
4. Road metaphor was liked and found useful for navigation through hierarchy of screens
5. Prefer drop down box to typing-in for word input


Please wait while I prepare some
things for you Ruli! This could
take a while


HOME
HELP
EXIT
Hello, Ruli! How do you want to create your joke?
My favourite jokes
Subjects
Any joke
Kinds of joke
Words
Topic shop
Joke factory
Keyboard shop
Joke junction
Theatre
Type shop
Joke library

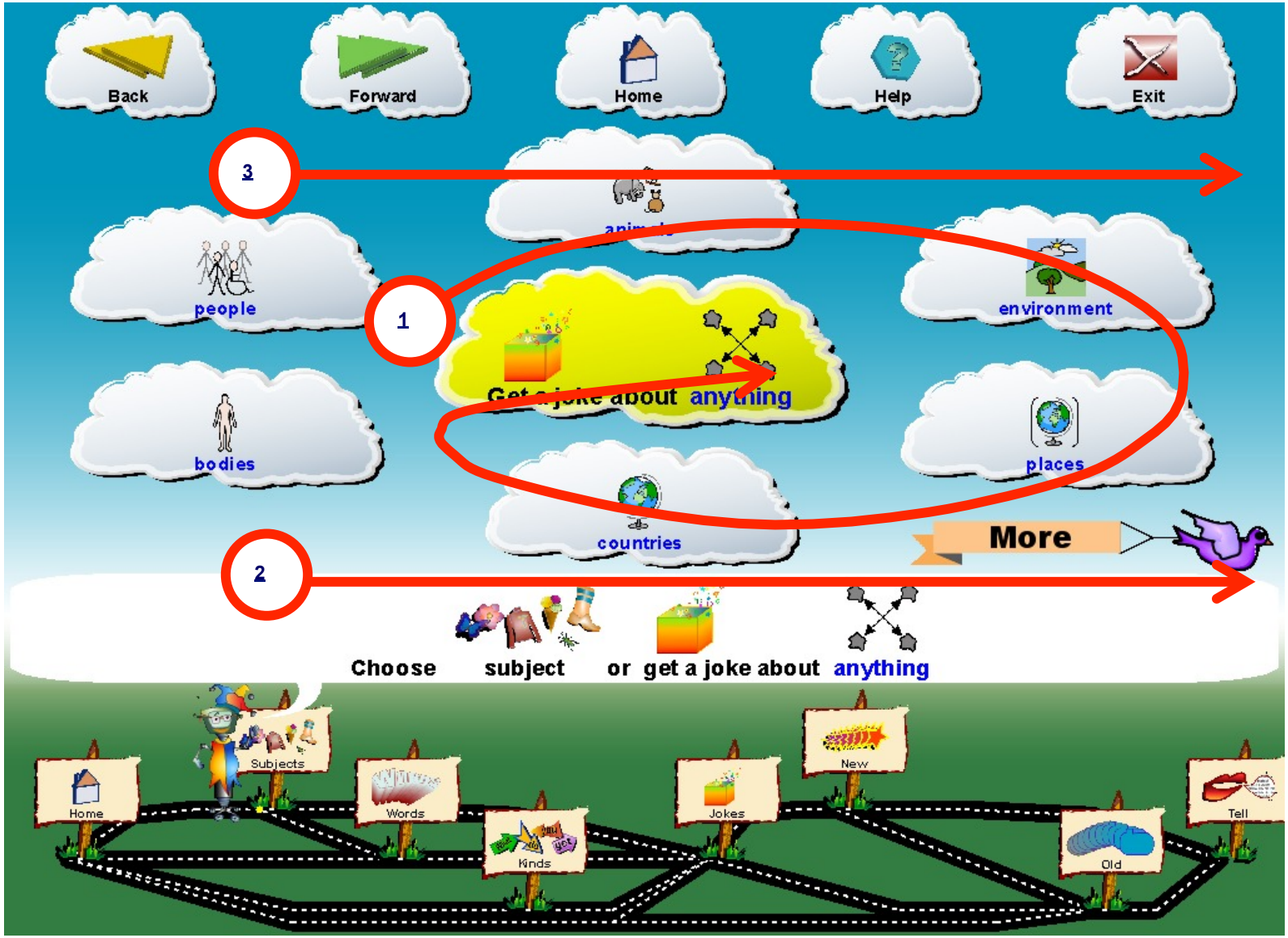
Later redesign

- The basic interface was redesigned by a graphic designer

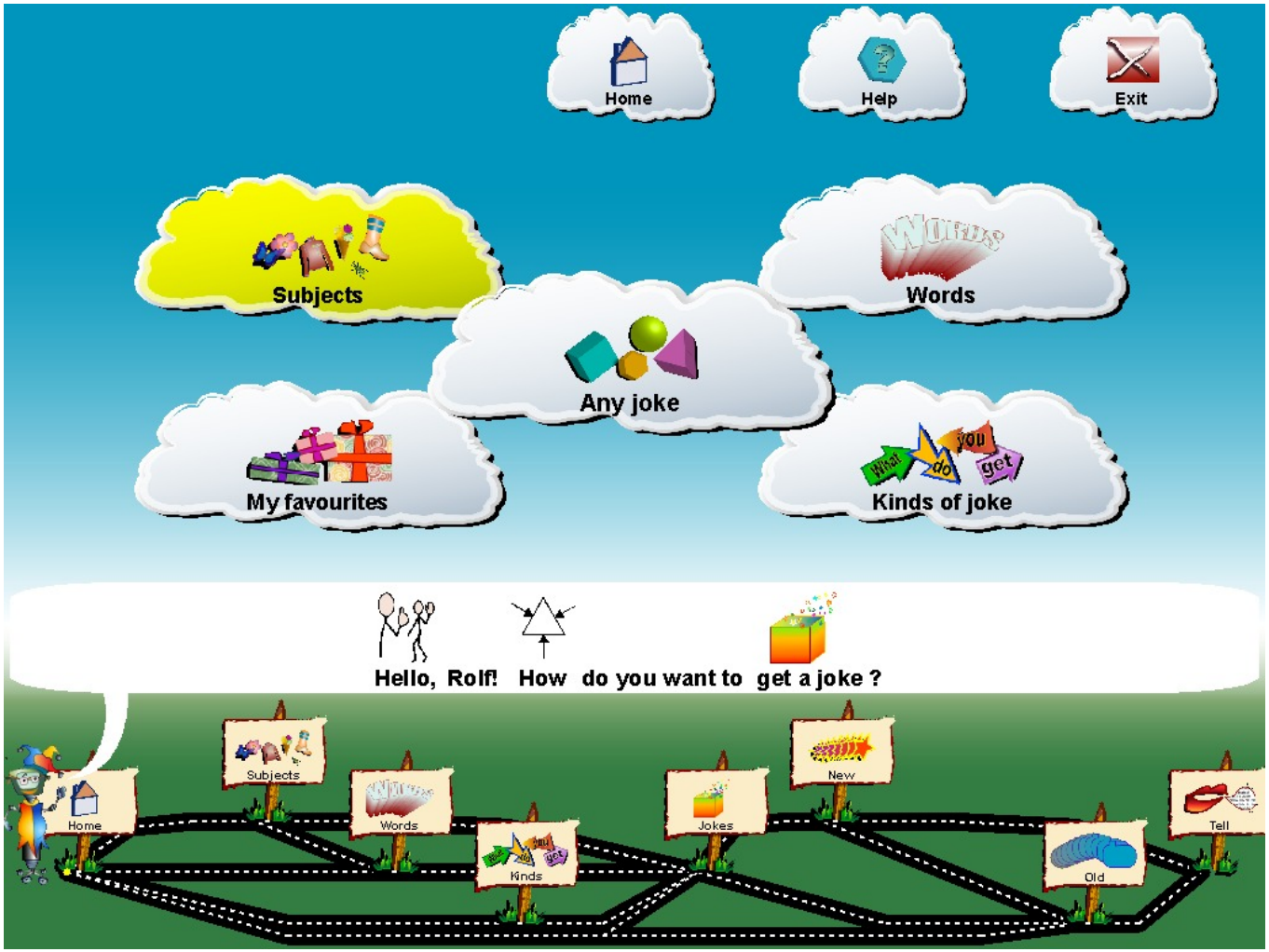
- Pilot tested with small group of typically developing children before use with target group

Note use of typically developing children, adults who were non-speaking children, and experts as *proxies* who stand in for the target population

Designing the Interface - Scanning



“Are you ready?” – Using STANDUP



3. Methods

Common Evaluation Methods

Task analysis
Observation
Cognitive Walkthrough
Mock-ups
Protocol analysis/think aloud
Wizard of Oz
Interview (structured/unstructured)
Questionnaire
Focus groups
Heuristic Evaluation
Expert evaluation
Sensitivity Analysis
Self Report
Post-hoc analysis
Logging use
Dialogue mark-up and analysis
Manipulation experiment
Sentient analysis

Direct Observation

Commonly used in **early stages of system design** or **hypothesis formation**

Identify potential **interactions between parameters** that might otherwise be missed

To help focus and record observations:

- use tools

e.g. *event counters, checklists, structured behavioural annotation sheets*

- restrict bandwidth

e.g. via *chat interface*

Very useful when used with other methods

Observation issues

Disadvantage: ***presence of the observer may affect behaviour being observed***

To reduce observer effects:

- **repeated sessions** enable participants to become accustomed to the observer's presence
- **careful placing of the observer** to avoid intrusion
- **train the observer** to resist interceding
- **explaining the role of the observer** to the participants

Mock-ups and paper prototypes

Goal: to get feedback on early design ideas before any commitment is made, mock-ups or prototypes of the system are used

- **electronic prototypes** can be developed and presented on computer screen
- **paper-based interface designs** can be used to represent different screen shots

*Elicits **responses to actual interfaces** and not other issues surrounding the operational access of technology*

*Facilitates **more imaginative feedback**, actively encourages **"hands on" interaction***

Video recording

Videoing user and system (or user and expert in WOZ studies) interaction **enables all visible user behaviour** (verbal and non-verbal) **to be used as data**

Video can be used for:

- **detailed behavioural analysis of user**
- in less detail, **for reference**, to determine interesting episodes in the interaction
- **to transcribe verbal interactions** between expert/tutor and student in WOZ studies

Video recording of screen interactions **also enables data capture of keyboard use and mouse movement**

Tools that permit replay of the interaction including all interface actions are becoming more common and reliable.

Interviews

Used **to elicit knowledge from a user by direct verbal questioning**, and can be:

1. **very structured**: pre-determined questions in specified order with little room for elaboration in responses
2. **semi-structured**: permits variation in order of coverage of questions, open-endedness in responses, flexibility in question selection and potential generation of new questions
3. **open-ended:** with few specific pre-determined questions and further question generation being determined by the previous response

Generally easy to administer and to respond to…

Interviews, contd.

Commonly used:

- for feedback on **interface design** and **usability**
- to determine **users feelings** and **attitudes**
- to determine **appropriate variables**
- post-session **to confirm other data** collected

Interviews versus questionnaires:

- conducted ***verbally*** rather than in ***written*** form
- suitable for ***eliciting*** a wider range of ***data*** which ***users may find difficult to elucidate*** in writing and without prompting
- interviews ***more objective*** than open-ended, unstructured feedback

Risk of respondent being influenced by questioner

Questionnaires

Present questions to be answered in **written form** and are **usually structured**

To determine:

- **user characteristics** e.g. demographic, goals, attitudes, preferences, traits
- **users task knowledge**

Used **as a means of expert evaluation**:

- in the **design stage** and later development cycles
- to **validate system behaviour**
- to **evaluate system behaviour**

  e.g. comparison with other systems or human performance

Heuristic Evaluation

Rule of thumb, guideline or general principle to guide or critique design decision

- useful ***in design stages***
- useful ***for evaluating prototypes, story boards***
- useful ***for evaluating full systems***

Flexible and cheap

May **use heuristics e.g. for usability**

Small number of **evaluators** e.g. 3 to 5 each ***note violations of heuristics and severity of problem***:

1. how common
2. how easy to overcome
3. one-off or persistent
4. how serious a problem

Evaluating Usability: Steps

1. Select a representative group of users
2. Decide which usability indicators to test (e.g. learnability, efficiency)
3. Decide the measurement criteria
4. Select a suitable test
5. Remember to test the software not the user
6. Collate and analyse data
7. Feed the results back into the product

Possible Usability Measures
(based on Waller, 2004)

1. The time users take to complete a specific task
2. The number of tasks that can be completed in a given time
3. The ratio between successful interactions and errors
4. The time spent recovering from errors
5. The number of user errors
6. The types of user errors
7. The number of features/commands utilised by users
8. The number of system features the user can remember in a debriefing after the test
9. The proportion of user statement during the test that were positive versus critical toward the system
10. The amount of 'dead time' during the session

Nielsen's Usability Heuristics

1. Visibility of system status
2. Match between system and real word
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and ease of use
8. Aesthetic and minimalist design
9. Help users recognise, diagnose and recover from errors
10. Help and documentation

Heuristic Evaluation: strengths and limitations (Waller, 2004)

- **Strengths**
  - Quick to perform
  - Relatively inexpensive
  - Uncover lots of potential usability defects

- **Limitations**
  - Several evaluations needed
  - Needs access to experts
  - "False alarm" risk
  - Serious *vs.* trivial problems
  - Highly specialised systems need highly specialised evaluators

Think Aloud/Protocol Analysis

User is recorded while talking through what he is doing

- ***what he believes is happening***
- ***why he takes an action***
- ***what he is trying to do***

Useful for design phase with mock-ups and observing how system is actually used

Advantages:

Simple, requires little expertise, can provide useful insights

Encourages criticism of system

Points of confusion can be clarified at time

Disadvantages:

But process itself can alter task

Analysis can be difficult

Possible Cognitive Overload

Logging Use

Automatic recording of user actions can be built into software for later analysis

- Enables replay of full interaction
- Keystroke and mouse movement
- Errors
- Timing and duration of tasks and sub-tasks

Advantages:

1. Objective data
2. Can identify frequent use of features
3. Automatic, and unobtrusive

Disadvantages:

1. Actions logged need to be interpreted
2. Technical problem and file storage
3. Privacy issues

Cognitive Walkthrough

User is asked to reflect on actions and decisions taken in performing a task, **post-task**

1. Re-enact task, replay session or use session transcript
2. User is asked questions at particular points of interest

Timing:

- ***immediately post-task*** (easier for user to remember)
- ***later*** (more time for evaluator to identify points of interest)

Useful when talk aloud would be too intrusive

Physiological Responses: Eye Tracking

Measure **how users feel** as well as what they do

Eye Tracking: now less invasive (not previously suitable for usability testing)

- Reflect amount of cognitive processing required for tasks
- Patterns of movement may suggest areas of screen that are easy/difficult to process

Can measure:

1. Number of fixations
2. Fixation duration
3. Scan path

Need more work on how to interpret, *e.g. if looking at text is user reading it?*

Becoming standard equipment

Physiological Responses: other measures

Emotional response may be measured through:

Heart activity - may indicate stress, anger

Sweat via **Galvanic skin response** (GSR) - higher emotional state, effort

Electrical activity in muscles (EMG) - task involvement

Electrical activity in brain (ECG) - decision making, motivation, attention

Other **stress measures**, e.g. pressure on mouse/keys

Exact relation between events and measures is not always clear

Offers possibly objective information in particular to inform affective state of user

References

Preece, J., Rogers, Y., Sharp, H., Benyon, D. Holland, S. and Carey, T. (1994). *Human-Computer Interaction*. Addison-Wesley

Dix, A., Finlay, J., Abowd, R. and Beale, R. (2004) *Human-Computer Interaction.* Prentice Hall

Lewis, C. and Rieman, J. (1994) *Task-Centered User Interface Design.* Shareware web publication, available at: http://hcibib.org/tcuid/

Meyer-Johnson. (2005). Picture Communication System (PCS) symbols are © Mayer Johnson Co., PO Box 1579, Solana Beach, CA 92075, USA.

Shneiderman B. (1998). Designing the user interface: Strategies for effective human computer interaction 3rd Ed. Addison-Wesley, Reading, MA.

Wood, L. (1998). *User interface design: Bridging the gap from user requirements to design*. (Florida: CRC Press).

References

Dix, A., Finlay, J., Abowd, R. and Beale, R. (2004) *Human-Computer Interaction.* Prentice Hall

Lewis, C. and Rieman, J. (1994) *Task-Centered User Interface Design.* Shareware web publication, available at: http://hcibib.org/tcuid/

Preece, J., Rogers, Y., Sharp, H., Benyon, D. Holland, S. and Carey, T. (1994). *Human-Computer Interaction*. Addison-Wesley

Meyer-Johnson. (2005). Picture Communication System (PCS) symbols are © Mayer Johnson Co., PO Box 1579, Solana Beach, CA 92075, USA.

Shneiderman B. (1998). Designing the user interface: Strategies for effective human computer interaction 3rd Ed. Addison-Wesley, Reading, MA.

Wood, L. (1998). *User interface design: Bridging the gap from user requirements to design*. (Florida: CRC Press).

STANDUP related references: see http://www.csd.abdn.ac.uk/research/standup/ and also http://www.csd.abdn.ac.uk/~gritchie/jokingcomputer/

Binsted, K. and Ritchie, G. (1994) An Implemented Model of Punning Riddles. Pp. 633-638 in *Proceedings of the Twelfth National Conference on Artificial Intelligence/*Sixth Conference on Innovative Applications of Artificial Intelligence (AAAI-94).

Binsted, K. and Ritchie, G. (1997). Computational rules for punning riddles. *HUMOR*,10 (1), pp.25-76

Low, A. (2003). *Software Support for Joke Creation*. 4th year project report, School of Informatics, University of Edinburgh, Edinburgh, UK.

Trujillo-Dennis, L. (2003). *An Accessible Interface for a Joke Creation Tool*, 4th year project report, School of Informatics, University of Edinburgh, Edinburgh, UK.

Waller, A., O' Mara, D., Manurung, R., Pain, H. and Ritchie, G. (2005) Facilitating User Feedback in the Design of a Novel Joke Generation System for People with Severe Communication Impairment. *Proceedings of HCI 2005* (to appear).

Further References

Cohen, P. (1995) *Empirical Methods for Artificial Intelligence*, MIT Press, 1995.

Conlon, T. and Pain, H. (1996). Persistent collaboration: a methodology for applied AIED, *Journal of Artificial Intelligence in Education,* 7, 219-252.

Conlon, T. (1999). Alternatives to Rules for Knowledge-based Modelling. *Instructional Science* Vol 27 No 6, pp 403-430.

Corbett, A.T. and Anderson, J.R., (1990) The Effect of Feedback Control on Learning to Program with the Lisp Tutor, *Proceedings of the 12th Annual Conference of the Cognitive Science Society*, LEA, New Jersey, 1990

Luger, G. F. and Stubblefield, W. A., (1989) *Artificial Intelligence and the Design of Expert Systems,* Benjamin Cummings, 1989.

Mark, M.A. and Greer, J.E. (1993). Evaluation methodologies for intelligent tutoring systems, *Journal of Artificial Intelligence in Education,* 4, 129-153.

Shute, V. J., & Regian, W. (1993). Principles for evaluating intelligent tutoring systems. Journal of Artificial Intelligence in Education, 4(2/3), 243-271.

Squires, D., & Preece, J. (1999). Predicting quality in educational software: Evaluating for learning, usability and the synergy between them. Interacting with Computers, 11(5), 467-483.