The Development of an Application to Enhance and Facilitate Collaborative Working within Groups in Higher Education

Paul Kiddie, Mike Sharples et al.
Motivation

- EECE students at the University of Birmingham currently use paper logbooks to:
  - Note down ideas
  - Record achievements, difficulties etc
- Disadvantages of the paper logbook in general:
  - Cannot add multimedia content
  - Written notes inflexible
- In groups:
  - Cannot effectively share the logbook between group members
  - Therefore, cannot effectively collaborate.
Aims

- To provide an integrated suite of tools to facilitate and enhance small group and collaborative learning within higher education
  - Provide existing tools on host machine via Interactive Logbook interface
  - Provide unique features such as location awareness of other students and adaptive access to teaching material
Socio-cognitive Engineering

- Breaks system design into building blocks.
- Ensures a User Centred approach to development.
- Iterative approach to system design, evaluation and redesign
Requirement Elicitation

- Students involved throughout design process, inherently by Socio-cognitive Engineering.

- Literature Review
  - Research into current Mobile Educational Technology

- Field Studies
  - Questionnaires gave a prioritised feature set
  - Focus Groups enabled gathering of a wider range of ideas.
Main Requirements

- Text communication to group members
- Collaborative document creation
- Group sharing of files and documents
- Group member location positioning
- Group and personal organiser
- Journal style application
- Integration with existing Microsoft applications
Implementation

Several technologies used:

- **C# .NET**
  - A powerful, Object Oriented language
  - UML methodology followed

- **Tablet PC SDK**
  - Easy interaction with XP Tablet PC edition
  - Level of support in online help

- **Microsoft SharePoint**
  - Ability to effectively share resources using the Internet.

- **XML**
  - Production of simple databases
  - User authentication, Appointments & Shared data locations
Ensuring Usability

- Ensure continuity between Interactive Logbook and other popular applications
  - Icons, Tool tips, Menus
  - Expected functionality when desired button pressed.

- Methods of evaluating
  - Draft interface illustrations
    - Interactive Logbook interface is adaptation of most popular
  - Card sorting
    - Students sorted functions into particular groups
  - Heuristic Evaluation
    - Identify obstructions to effective use, such as bugs and crashes.
This part of the screen will allow the launch of instances of apps such as internet explorer, powerpoint viewer, pdf viewer, etc...

Member Locations

<table>
<thead>
<tr>
<th>John Smith</th>
<th>Room 23</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paul Jones</td>
<td>Room 23</td>
</tr>
<tr>
<td>Peter Davies</td>
<td>Room 06</td>
</tr>
</tbody>
</table>

This Represents Floor Layout of the Building
where $K_1 + K_2 + K_3 + K_4$ are information bits.

Example:

Connect the Hamming (7,4) code word.

$C_2$ is an error.
The Launch Bar

- Provides the main interactions between student and Interactive Logbook
  - ‘Programs’ provides access to applications or external facilities
  - ‘Modules’ provides a teaching modules’ specific material
  - ‘Meeting’ provides collaborative tools
  - ‘Diary’ provides time management facilities
The Hamming (7,4) Code

- An example of an \((n,k)\) linear block code. Each codeword contains \(n\) bits, \(k\) information bits and \((n-k)\) check bits.
- Hamming (7,4) is a nice easy code but is not very efficient (a 75% overhead). It generates codewords with a 'Hamming distance' of 3, i.e., all codewords differ in 3 locations. It can correct one bit in error and detect two.

In the codeword \([C] = k_1 \ k_2 \ k_3 \ k_4 \ c_1 \ c_2 \ c_3\), \(k_1-k_4\) are information bits and \(c_1-c_3\) are check bits (the 'systematic redundancy') are calculated as:

\[
\begin{align*}
  c_1 &= k_1 + k_2 + k_4 \\
  c_2 &= k_1 + k_3 + k_4 \\
  c_3 &= k_2 + k_3 + k_4
\end{align*}
\]

So the data \([X] = 0 \ 1 \ 1 \ 0\) becomes the codeword

\[
[0 \ 1 \ 1 \ 0 \ (0+1+0) \ (0+1+0) \ (1+1+0)] = [0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0]
\]
Notes/Whiteboard session

- Tablet PC
- Mobile Learning
- Mobile Phone
- PDA
- Laptop PC
Administration

- Only preliminary support
- Three main areas of administration
  - Utilisation of Interactive Logbook on any shared network storage
  - Addition of teaching material and other resources to their timetable entries
  - Creation of year timetables with the above detail in place to offer to students using Interactive Logbook.
- Ensures adaptability to any Higher Education environment.
Future Developments

- Improvements to reliability of existing functionality
  - Robust messaging
  - Sophisticated data handling using SQL
- Extensive use of wireless location awareness by Ekahau to adapt Interactive Logbook interface to location in department
- Transition onto smaller mobile devices with thin Java client.
- Addition of multimedia content onto inked notes
- An ‘open’ architecture to enable the uptake by other HE institutes who may use different technologies, e.g. web shares (SharePoint, WebCT)
Conclusions

- Interactive Logbook is a tool to facilitate and enhance small group and collaborative learning within higher education.

- Its benefits to students are:
  - Increasing of collaborative possibilities
  - Increasing productivity of small groups
  - Increasing communication

- Necessity to develop for the user in mind

- Still a prototype product
  - Needs development before Interactive Logbook can be deployed.
interactive logbook

Thanks for listening!

http://www.interactivelogbook.tk