



Annotating Digital Documents

1



Ink Annotations are different to text annotations

- Ink stands out from the original
- It is free form – the annotator can
 - Emphasize – underline, highlight, asterisk
 - Question
 - Agree
 - Add side notes to explain, clarify
- The authorship remains with the original author

2



Why do we annotate?

- Active reading
 - Adding annotations to a document helps the reader to process the information
- Communicate with others/self
 - When developing the document collaboratively
 - When evaluating the work (ie marking assignments)

3



Expressiveness and usefulness

- Annotations are very individualistic
- However they can generally be understood by others.
- Interesting studies of students choosing old – already annotated books – and looking carefully at the style of annotations
- Also students understanding pre-annotated material more quickly than the original document

4



Visual message

- Looking at an annotated document in an instant you understand the position of the annotator. Are they:
 - Being critical?
 - Correcting the spelling and grammar?
 - Adding their own explanatory notes

5



Example Software

- Penmarked
 - Assignment marking
- RCA
 - Code review

6



Penmarked Motivation

- Electronic submission and marking of assignments is attractive to faculty and students alike
- Particularly programs assignments when a digital copy of the assignment is required for marking
- However providing meaningful feedback to students is more difficult
- And individualised feedback is important

7

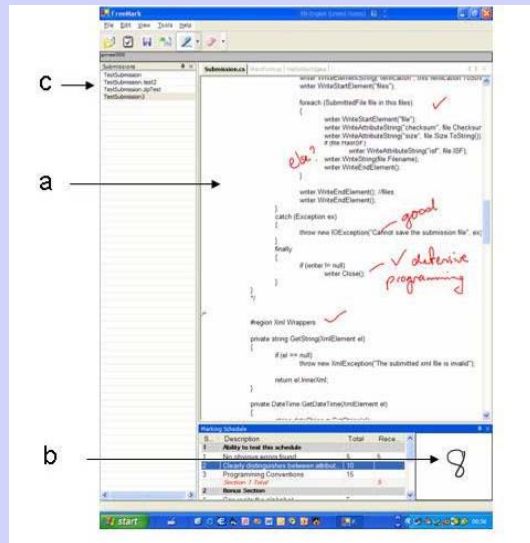


Design Brief

- Annotation of the script
- Score recording
- Work practice support
 - Who wants to 'waste time' marking?

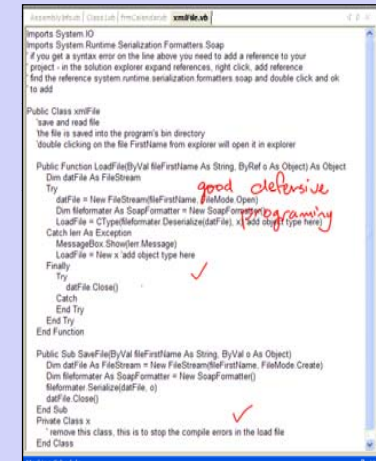
8

Implementation

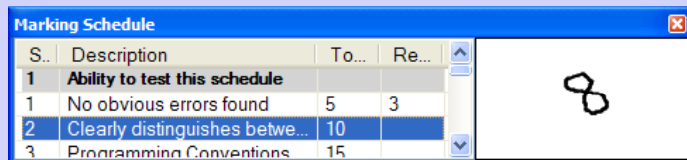


Annotation Pane

- Ink anywhere
- Erase
- Multiple files
- A technical nightmare
 - Two layers
 - Lots of Windows API calls



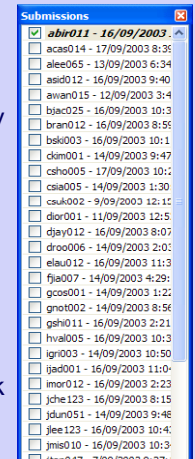
Scoring Pane



- Marking rubric
- Numeric scores
- Entered
 - in score box with pen and recognized using Microsoft recognition engine with numeric factoid feature also range checked
 - Or through the keyboard

Work Practices Support

- Pre-marking
 - Set up of marking rubric, directories, file type/name filters
 - Collection of assignments from a dropbox or directory structure
- During
 - All assignments listed in student list pane
 - Can be marked as opened and complete
 - Automatic unpackaging of zip files
 - Direct link to student's folder for executing program
- On completion
 - Exporting of marks as xml file
 - Conversion of assignments to pdf files (includes mark rubric and annotated files)
 - Email return of assignment to student



Evaluation

- Iterative informal during development
- Usability testing with think aloud
 - Talking while marking a program is challenging
 - A few interaction problems
 - pen/keyboard input
 - 'do you want to save' dialogue
 - Recognition required correct formation of digits
 - Using the pen to interact with non-tablet programs (the students assignments) is slow!
- Focus group
 - Student list and workflow support invaluable
 - Accurate writing required for score recognition

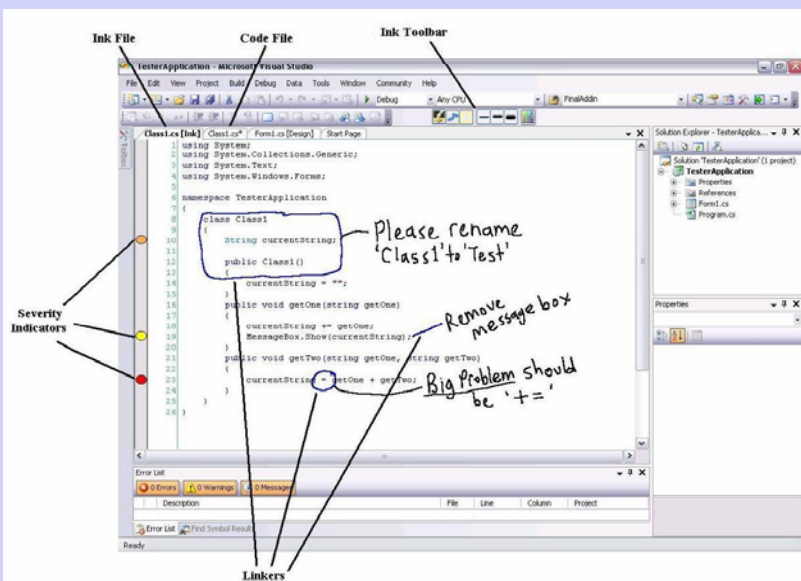
13

RCA

- One of the difficulties with Penmarked is that the markers have to go to another application to compile and execute the program
- RCA is an annotation addin for Visual Studio

14

RCA in Visual Studio



Technically

- Transparent overlay
 - Another technical nightmare!
- Each annotation consists of:
 - A linker – line or circle
 - Annotation – group text or diagram
 - A severity indicator

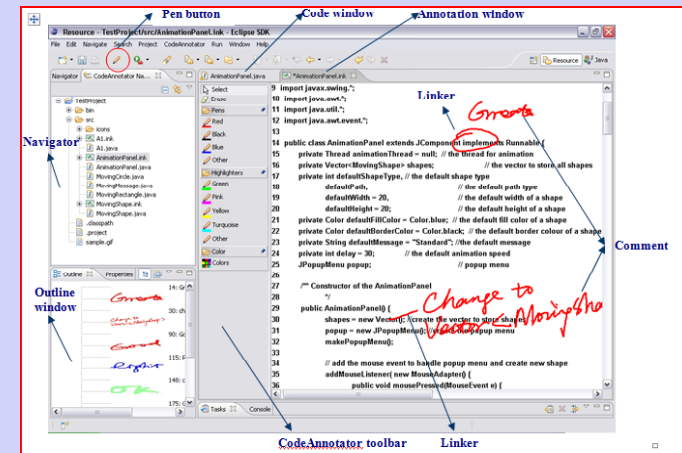
16

CodeAnnotator

- Same idea but trying to use Eclipse.
- Added some navigation and character recognition
- Pretty much the same problem!
- No extensibility points from the code windows!

17

CodeAnnotator UI



18

Related Work

- Paperless environments
 - Sellen and Harper (2002)
- Ink interaction
 - Plimmer and Apperley (2003)
 - Freeform - UI sketching environment
 - Jarrett and Su (2003)
- Annotation
 - Marshall (1997)
 - Shipman, Price et al (2003)
 - Wolfe (2000)
- Recognition
 - Tablet OS
- Marking software
 - Heinrich and Lawn (2004)

19

Discussion

- Why is it so difficult?
 - There is a fundamental digital divide between text and images
 - Trying to combine the two and keep them consistent is really difficult
 - Knowing what to do with ink when the underlying document changes its problematic
 - Many standard window do not easily support non-text

20



Future Work

- Trying again in Eclipse in the eclipse base code (maybe)
- Annotation in web browsers.

21



Finally

- Going paperless requires
 - Providing informal inking
 - Reliable recognition (sometimes)
- and
 - Attending to work practices which are more varied than we first imagine!

22