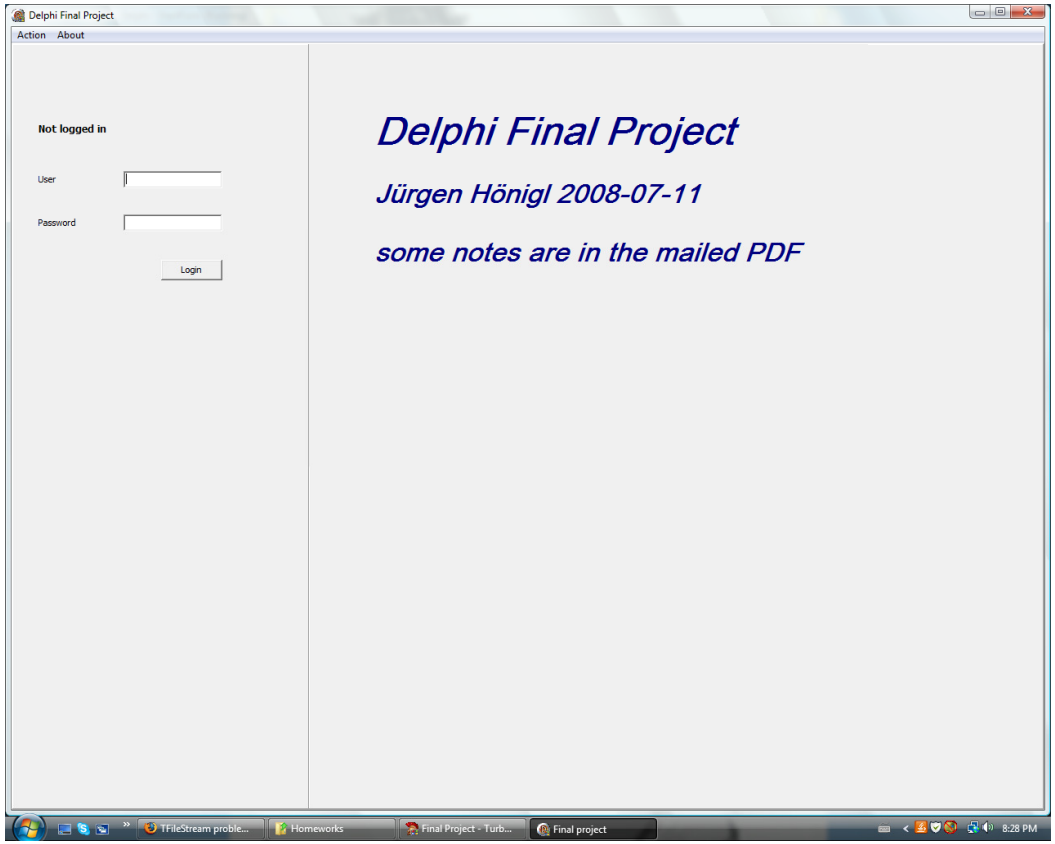
# Delphi Final Project

some notes about my project




Jürgen Hönigl

# For testing (pre-defined accounts)

- User: **0355661**Pass: **65,,,,** Role: **Student**
- User: **p99222** Pass: **62,,,,** Role: **Teacher**
- User: **p77661** Pass: **40,,,,** Role: **Admin**

# Account description

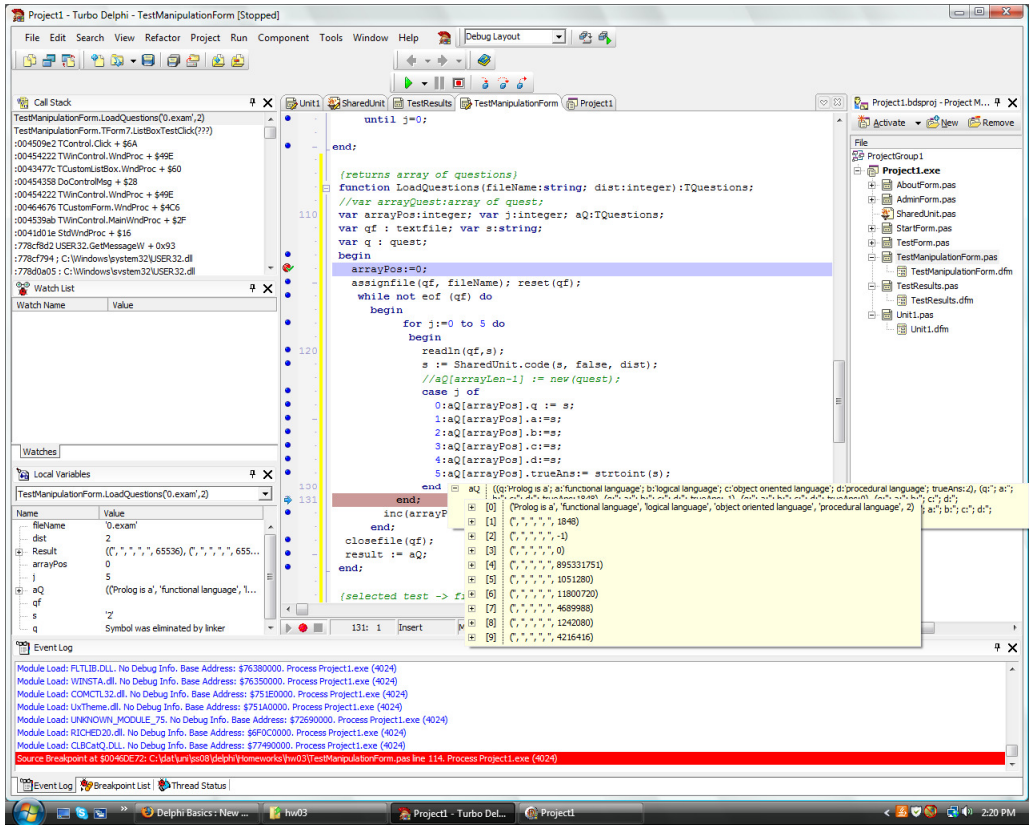
- Example: KUSSS Login(for students):
  - 0355661
    - 03 -> 2003
    - 55 -> JKU Linz
    - 661 -> Student number
- -p99222 (my assumption/design)
  - P -> Personnel
  - 99222 -> Personnel number

# ZIP

- contains Project file (Borland Turbo Delphi 10)
- contains .pas source
- contains .dfm forms

# Development

- Tested with MS Windows Vista Business
- IDE : Borland Turbo Delphi 10



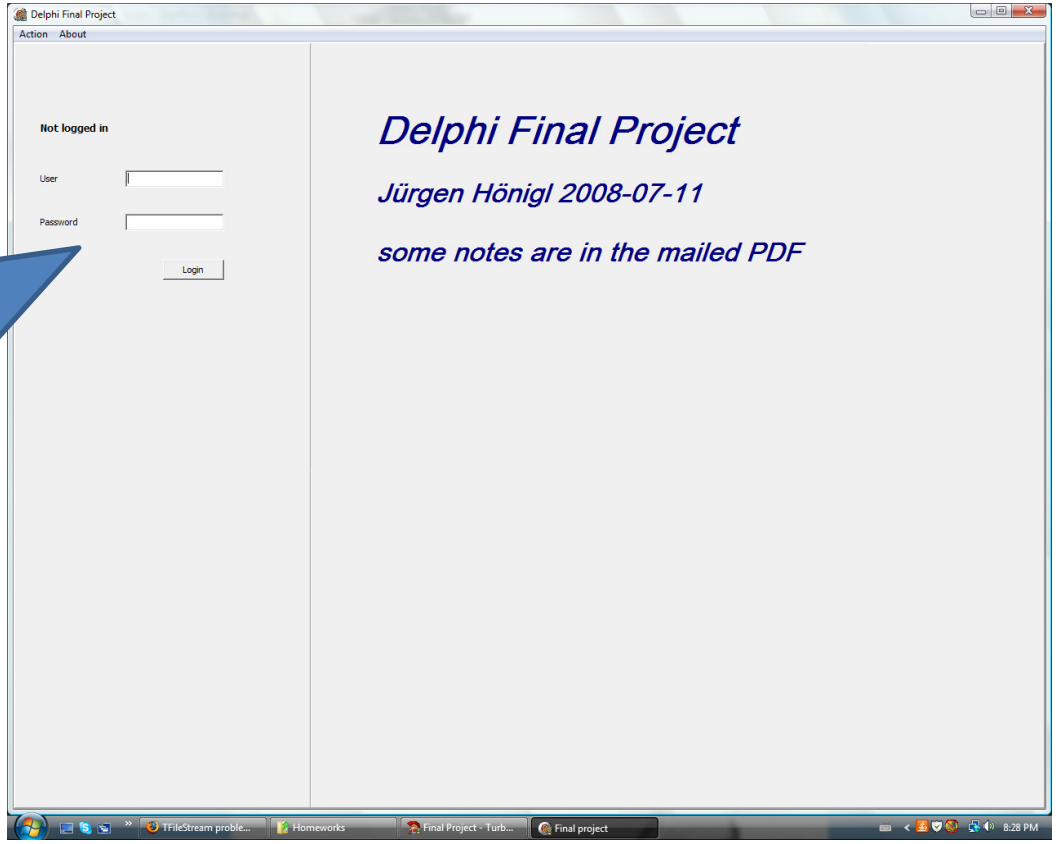
# Logins

- Wrong user and/or wrong password
  - Only feedback -> ‘Not logged in’
- Because feedback such as
  - User does not exist
  - Wrong password

are only hints for testers (users without an account)

# Screenshot

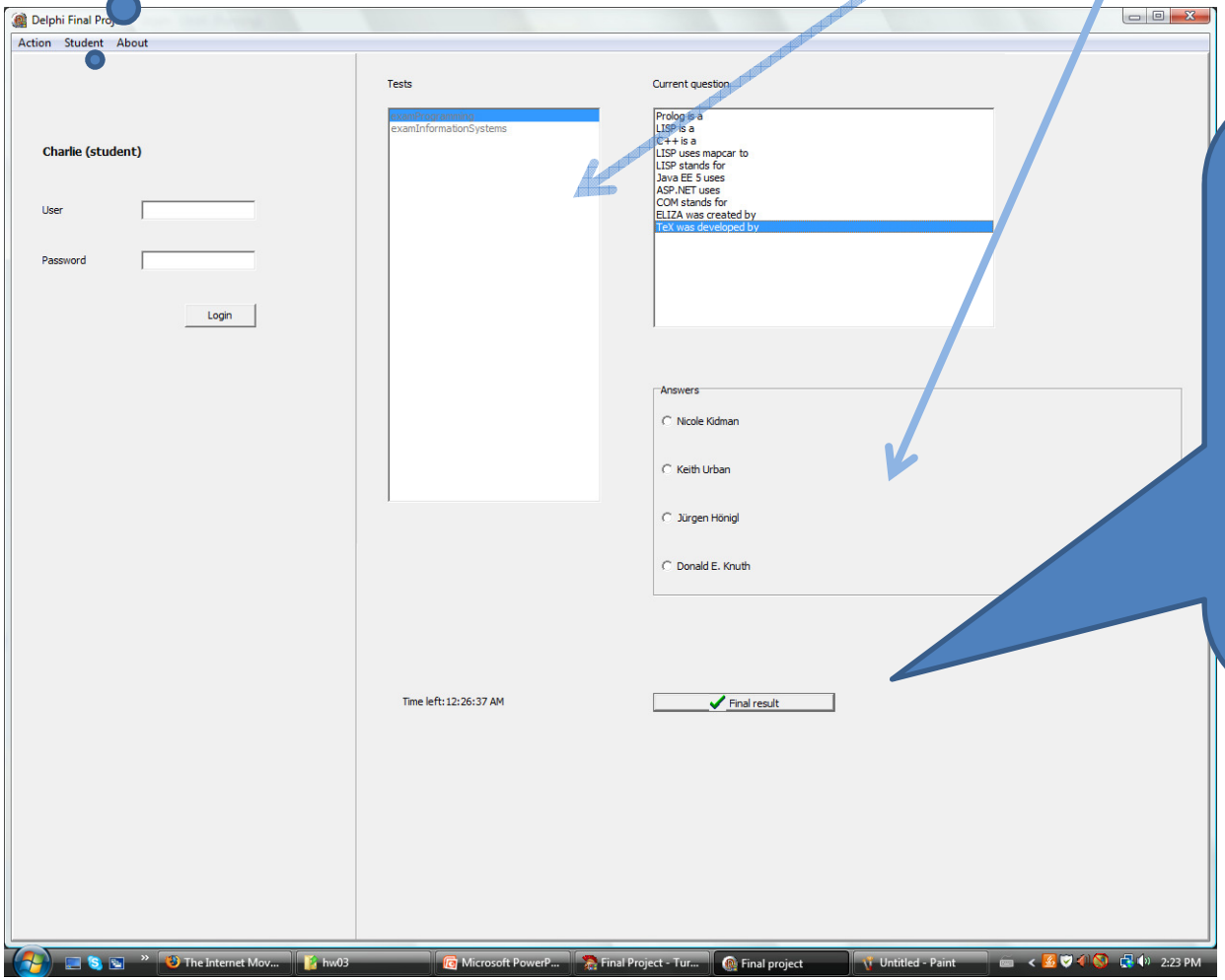
Login
Student/Teacher/Admin



Menu
Student

# Screenshot (2)

selection of a test -> ListBoxTest disabled to focus on questions and answers

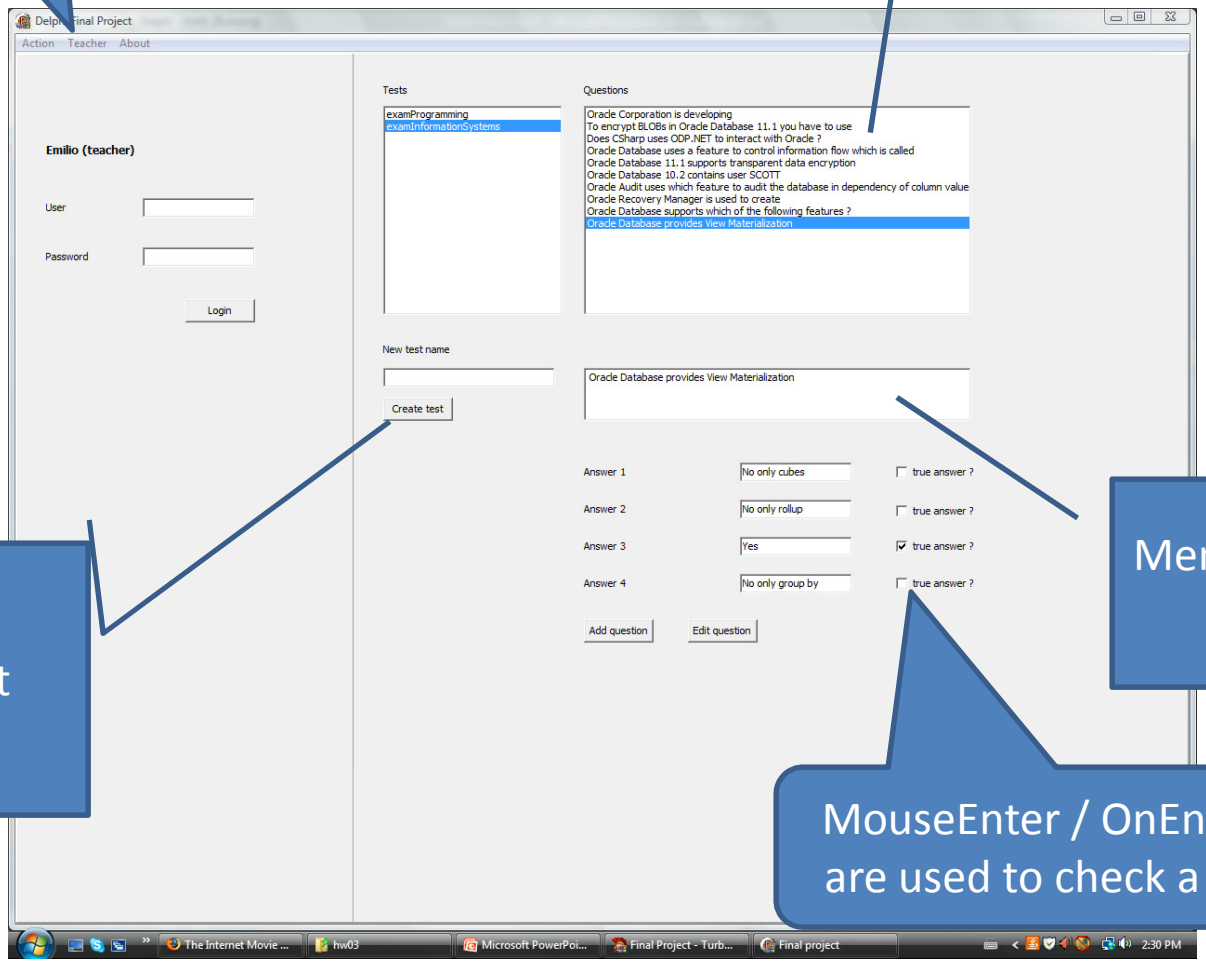



Student has permission to abort a test(MainMenu Logout) – Student get only a result if he/she finish a test

# Screenshot (3)

Menu Teacher

10 Questions / Test



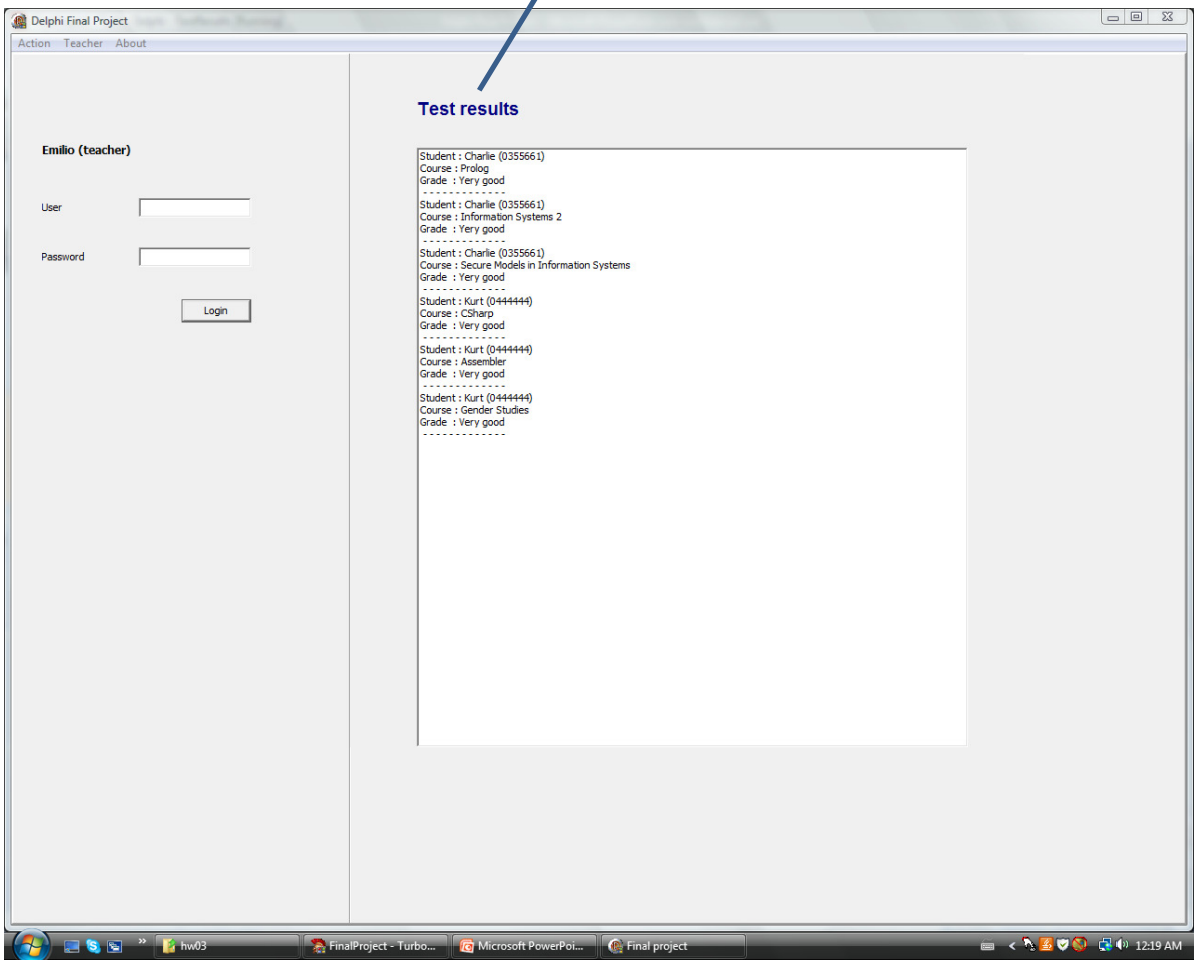

Create a test

Memo for add/edit a question

MouseEnter / OnEnter events are used to check a CheckBox

# Screenshot (4)

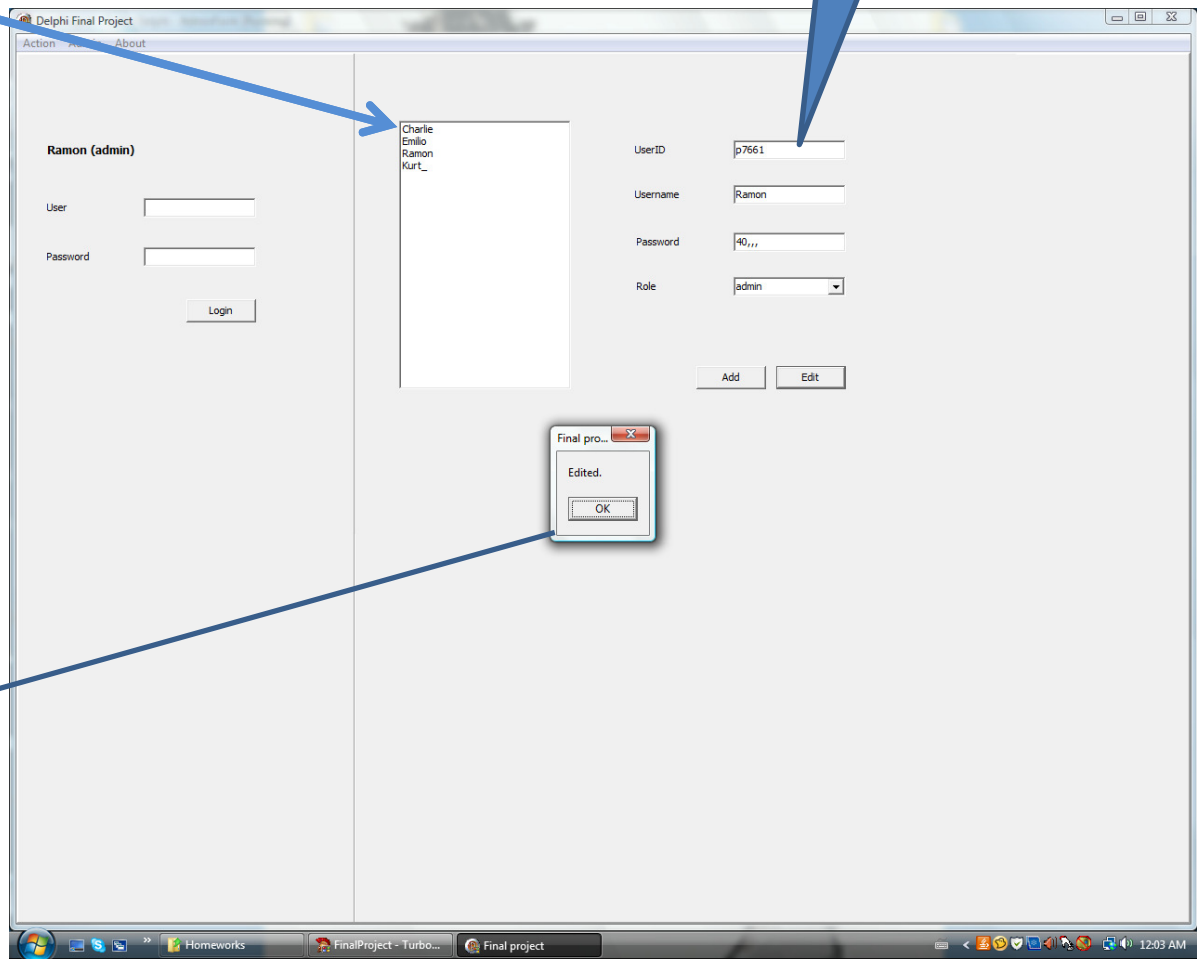
Teacher – Observing all results



# Screenshot (5)

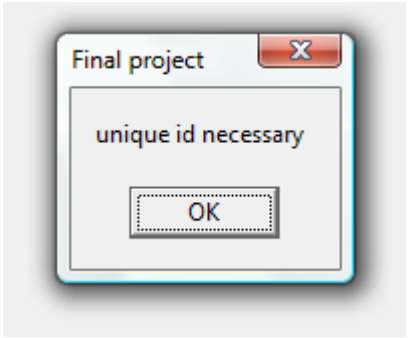
Admins
Students
Teachers

EditBox ID only for adding

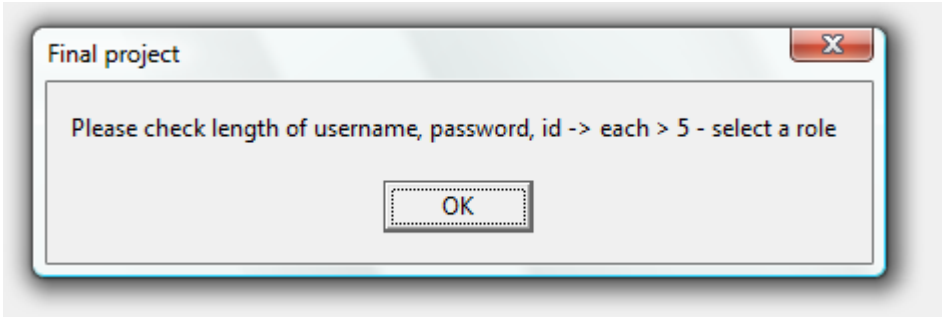



ShowMessage

# Screenshot (6)

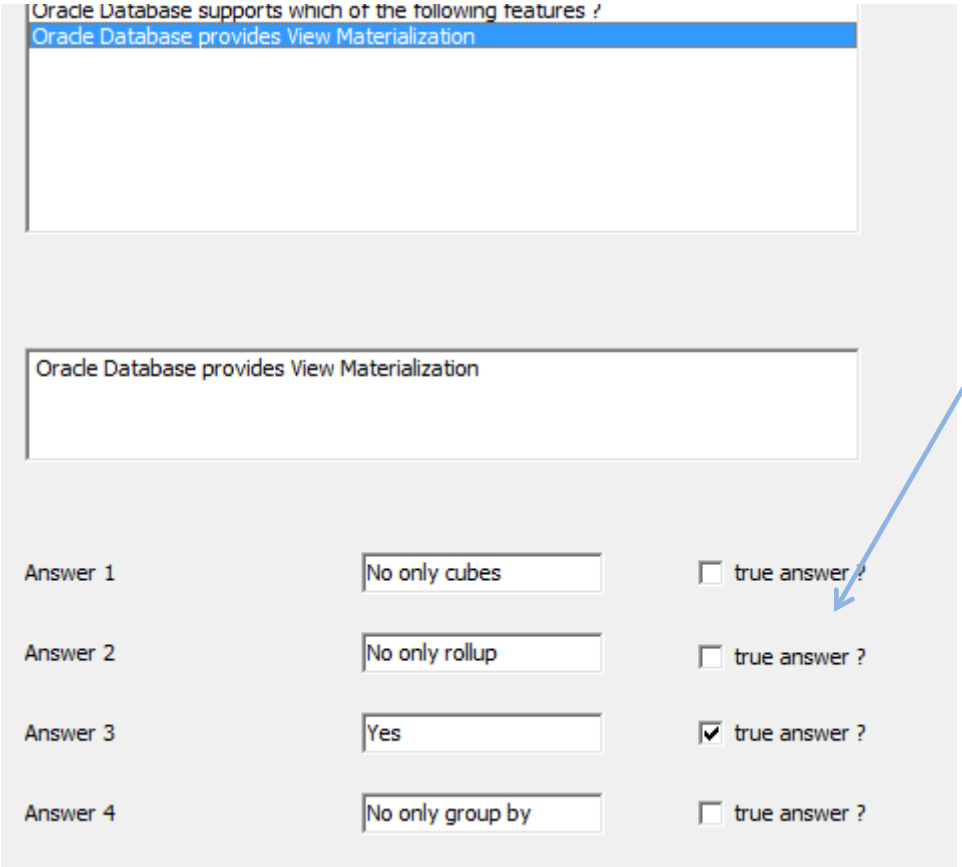



Unique ID will be checked only for adding – Edit can not edit an ID (AdminForm)




Length will be checked for adding/editing (AdminForm)

# Screenshot (7)




No mouse click necessary – MouseEnter / OnEnter event are used.

# Code

- Some shared code such as
  - **Code (for decode or encode)**
  - **LoadData**

  is in the unit **SharedUnit**
- Avoiding redundancy

# Code (2)

- Form **TestResults** is used for
  - assignment **1e (‘ALL’)**
  - assignment **2d (e.g. ‘0444444’)**
- Avoiding redundancy

# Code (3)

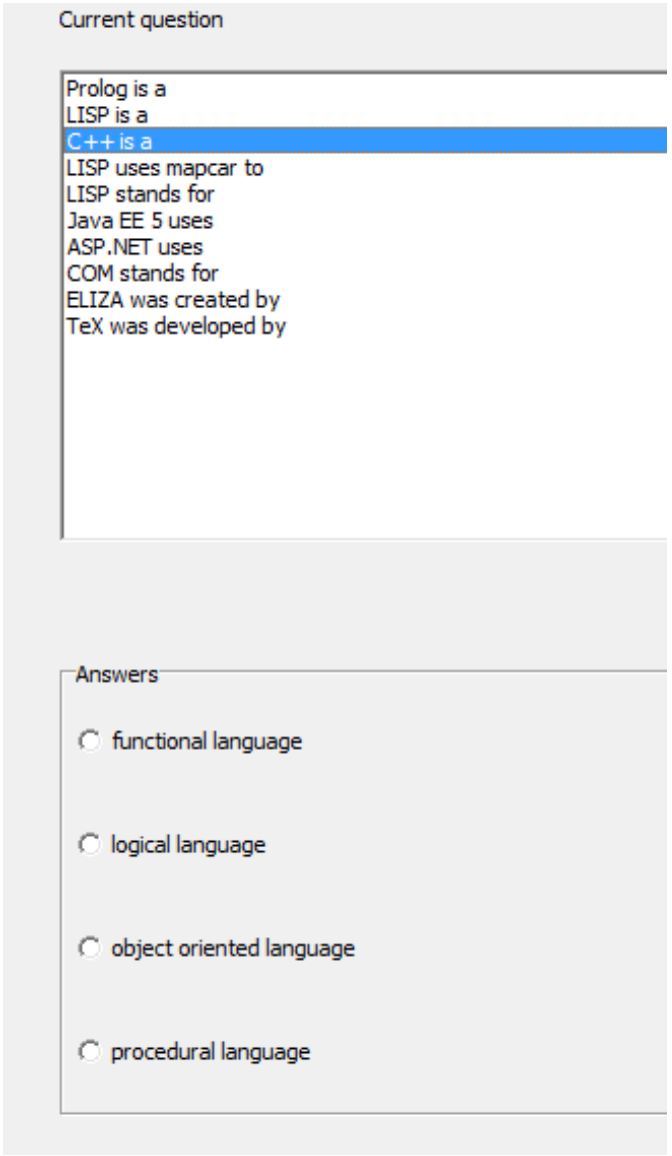
- TestForm
  - based on your example

# Test

- Contains 10 questions
  - each question has 4 possible answers
  - only one answer is correct
- Tests with 3 or 4 questions can also be calculated
  - Percentage

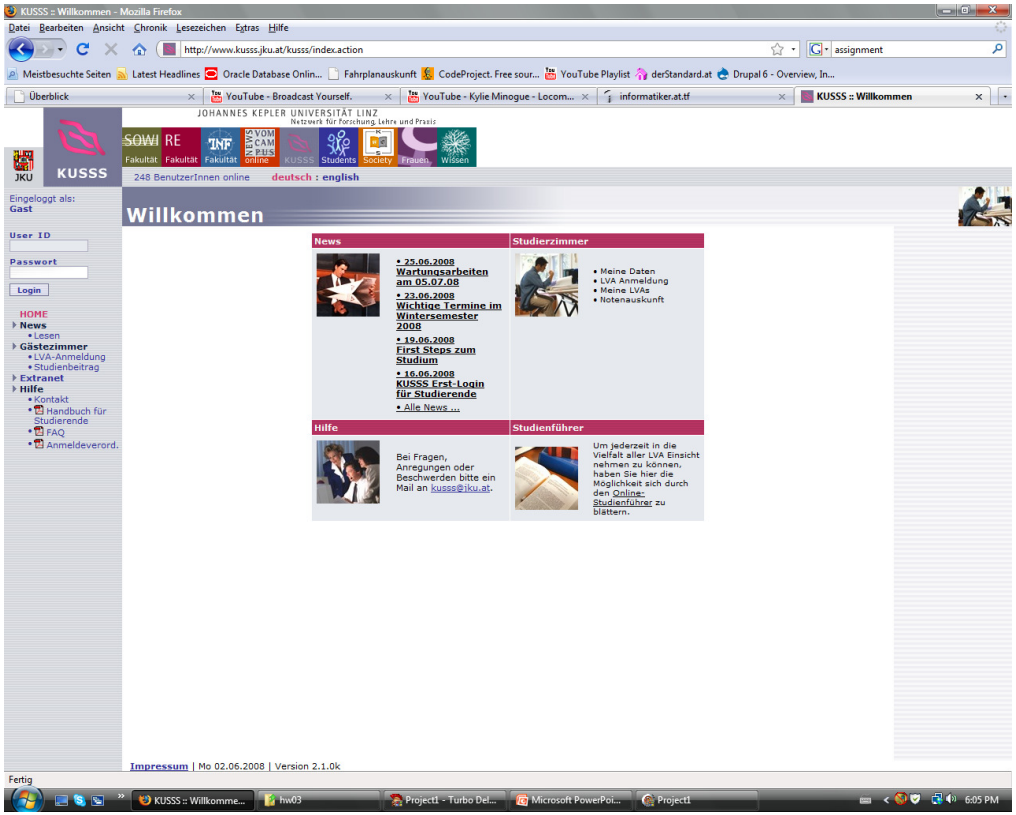
# Test (2)

- Question 1-3 in programming test have same possible answers (and a different true answer) because questions are similiar (*LanguageXY is a*)



# GUI

- Login -> a little bit inspired by KUSSS




Picture : Screenshot KUSSS

# GUI (2)

- One MainForm
  - hosts another forms in a panel
  - using Parent property
  - using Visible property / Show

# Roles

- Teacher
- Student
- Admin

# Test results

- Student
  - can see his/her results
- Teacher
  - can see all results of students
- Admin
  - can not see any results of students

# Data in files

- Using encoded data
- UserData stored in file such as
  - UserID(unique)|UserName|Password|Role

# Actions

- Login
  - In left panel
- Logout
  - Menu: Action / Logout

# Actions (2)

- Student
  - see menu Student
  - after successful Login
  - actions from assignment

# Actions (3)

- Teacher
  - see menu Teacher
  - after successful Login
  - actions from assignment

# Actions (4)

- Admin
  - see menu Admin
  - after successful Login
  - actions : AddUser && EditUser

EOP