

Gecracktes MP4-gewinnt im Multiplayer mit Neuigkeiten

Was soll das bitte?

Sechs Teams haben sich bei mir angemeldet. Das heisst, die dazugehörigen Leute bekommen auch ihr Testat, wenn sie die Projekte abschliessen:

MP3 in Ordnung bringen *David Murer, Timon Hertli, Marko Ristin*

Ein Programm, das eine geordnete Datenbank für wild genannte MP3-Dateien erstellt. Sortierte Alben, welche mit statistischen Methoden zusammengefasst werden, sind das Endergebnis.

Vier-gewinnt *Marco Willi und David Bossert*

Das klassische Vier-gewinnt

X-gewinnt *Andreas Pasternak, Jonas Renggli, Peter Bucher, Simon Inderbitzin*

Ein Spiel auf der Basis von Vier-gewinnt, nur dass es auf einem beliebig grossen Feld x Steine in einer Reihe zu positionieren gilt.

Multiplayer Flathunt *Curd Bänziger, Stefan Eigenmann, Matthias Geel, Quoc Huy Nguyen*

Flathund kann im Multiplayer-Modus im Netzwerk gespielt werden.

EiffelRSS *Thomas Weibel, Michael Käser, Martin Luder*

EiffelRSS is an Eiffel library to parse and write RSS. The goal is to provide the Eiffel development community with an easy to use and good

structured API for RSS. It also contains a simple RSS newsfeed reader written with EiffelVision. Homepage: <http://eiffelrss.berlios.de>

DeCSS für Eiffel *Tomas Carnecky*

Das bestehende DeCSS Programm wird über C++ Externals an Eiffel angebunden.



Gil Carpalhoso, [home.fmh.utl.pt/~al07835/ images/snowboard.JPG](http://home.fmh.utl.pt/~al07835/images/snowboard.JPG), 500 x 338 Pixel - 26k

Volles Programm

Die Agenda ist dicht gedrängt, aber die Zeit sollte eigentlich locker ausreichen, wenn jeder mithilft:

Montag, 10. Januar

Ich erzähl etwas über GUIs allgemein und in Eiffel

Dienstag, 11. Januar

Ihr gebt mir ein verbindliches Pflichtenheft mit dem Aufbau eures Programmes (Skizzen von Fenstern, ein Musterprogrammablauf als Comic, Storyboard oder so)

Montag, 17. Januar

Ich erzähl euch, wie man testet.

Dienstag, 18. Januar

Ihr gebt mir euren Entwurf und die genaue Struktur eures Projektes ab: Klassenhierarchie, wichtige Features etc.

Montag, 24. Januar

Ich erzähl euch, wie die Dokumentation aussehen sollte.

Dienstag, 25. Januar

Jeder bringt sein Projekt auf dem Laptop und lässt es von einem anderen Team auf einem anderen Rechner laufen- testen, testen, testen...

Montag, 31. Januar

Ihr präsentiert die Projekte, zeigt mir die Dokumentation und den Quellcode. Jede Gruppe hat etwa 30 Minuten Zeit für wunderbare Folien, Gekrizel an der Wandtafel oder Vorführaktionen am Laptop.

Phasen, die euer Projekt durchmacht

Auf der Homepage seht ihr den Status eures Projektes. Folgende Reihenfolge von Phasen ist sinnvoll:

1. Projektbeschreibung
2. Entwurfsphase: Klassenhierarchie, Struktur, wichtige Features, etc.
3. Implementationsphase: Klassentext wird geschrieben, alle Features implementiert und vorzu zum kurzen Testen ausgeführt.
4. Testphase: das Programm ist soweit fertig, dass man es einigermassen einsetzen kann- bei Fehlern wird nochmals bei Phase 3 (Implementationsphase) weitergefahren.

5. Falls keine Fehler mehr da sind, werden die Erweiterungsmöglichkeiten eingebaut, und jeweils bei Phase 3 weitergefahren.
6. Wenn alles fertig ist, oder der Termin gekommen ist, dann wird präsentiert.

Während der ganzen Zeit schreibt ihr Kommentare im Quellcode (die kann man dann nur ganz mühsam später wieder reinaddieren). Damit euer Projekt von der Informatikzunft respektiert wird, müssen diese Kommentare professionel und hilfreich sein. Die anderen Dokumente werden vorzu verfasst.

Wie sieht eine Projektbeschreibung aus?

Damit ihr den Status Projektbeschreibung erhält, müsst ihr eine solche mir mailen. Eine Projektbeschreibung besteht aus

- Projekttitle, fett als Überschrift
- Namen der Mitarbeitenden
- Management Summary: Funktionsweise (ähnlich einem Werbetext) oder bei einem Spiel das Spielprinzip
- Funktionalitätsmerkmale: ein grob abgefasstes Flussdiagramm, oder Abschnitte mit Untertitel Programmstart, mögliche Interaktionen, Programmende, Spielbeginn, Spielende, etc.
- Erweiterungsmöglichkeiten: das Programm soll noch beliebig erweitert werden können. Macht euch Gedanken dazu, damit ihr dann gegebenenfalls das auch noch implementieren könnt.

Wenn Ihr Fragen habt, dann mailt mir bitte unbedingt. Wir wollen keine Zeit verlieren.