

# Inoffizielle Modulbeschreibungen

Was die HSLU-Broschüren verschweigen

Patrick Bucher

08.06.2019

«*Das Schicksal ist grausam und die Menschen sind erbärmlich*»

— Arthur Schopenhauer, *Aphorismen zur Lebensweisheit*

- AD: *Algorithmen und Datenstrukturen* (und Nebenläufigkeit!) sagt schon alles. Es geht wirklich um Algorithmen und um Datenstrukturen (und um Nebenläufigkeit!), was anhand von mehr oder weniger sinnvollen Aufgaben eingeübt wird. Jede Woche werden drei von fünf vorher bestimmten Studenten ausgelost, die ihre Lösungen präsentieren müssen. Dabei kann es sein, dass manche Studenten *immer* und andere *nie* präsentieren müssen. (Siehe auch die Ausführungen zum Thema Zufall in der Beschreibung des Moduls PTA.) Von tieferen Semestern ist zu vernehmen, dass der Präsentierende aufgrund der gestiegenen Studierendenzahl neu mit gleich *zwei* Würfeln ausgelost werde. Bis zur Einführung des Modulo-Operators im Folgsemester müssen sich also Studierende, die sich auf der Liste neben der Zahl 1 wiederfinden, noch etwas mit dem Präsentieren gedulden.
- CAB: *Cloud Angebot und Betrieb*, siehe CI: *Cloud Infrastructure*
- CI: *Cloud Infrastructure*, siehe CAB: *Cloud Angebot und Betrieb*
- CNA: Computer- und Netzwerkarchitektur ist ein Sammelsurium von Themen. Die Vorlesung wird gehalten vom Martin-Schulz-Double Wannemacher, vom Systemtechnik-Schergewicht Waldmann und vom Professor für extraterrestrischen Mittelwellenfunk Klaper. Von Holzcomputer bis Mainframe, von Satellitenkommunikation bis ZigBee und durch alle OSI-Layer hindurch erhält der Studierende eine Schnellbleiche über *alles das Informatische Betreffende und alles damit Zusammenhängende*.

- CSA: *C# in Action* ist eine Einführung in die Programmiersprache C#, die von C#-Apoleten immer als «viel besser als Java» bewertet wird. Nach erfolgter Einführung soll ein zweimotoriger Roboter angesteuert werden, was gemäss studentischen Berichten immer aufgrund schlechter Konnektivität scheiterte. Höhepunkt des Moduls – aus Sicht des Autors dieser Zeilen – war zweifelsfrei die Vorlesung, die mit einer ausgiebigen Instruktion darüber begann, wie die Robotersoftware daran gehindert werden kann, Visual Studio zum Abstürzen zu bringen – und aus Sicht des Autors damit endete, dass er entnervt seine Sachen packte, dem Sonnenuntergang entgegen nach Hause ging, und sich nie wieder in einer Vorlesung zu diesem Modul blicken liess.
- DBS: Im Modul *Datenbanksysteme* machen wir *Big Data* mit cloud-basierten *NoSQL*-Datenbanken, die als *Docker*-Images auf einem verteilten *IoT-Kubernetes*-Cluster laufen, mit *Blockchain*, die wir per verteiltem Echtzeit-Java manipulieren, per *Machine Learning* trainieren und mittels *Augmented Reality* ... *BINGO!* Im Rahmen einer Satiresitzung lehnte sich der Modulverantwortliche Prof. Dr. Michael Kaufmann mit der kühnen These, dass Digitalisierung ein wichtiger Trend sei, weit aus dem Fenster und erwähnte auch den Begriff *Industrie 4.0*. Es werden unter anderem Legacy-Technologien wie SQL und relationale Datenbanken eingeführt, über die der Modulverantwortliche schon längst hinweg ist. Unter der Maxime «NoSQL, bitch!» werden zu relationalen Datenbanken gelöste Aufgaben als *zu sehr im relationalen Paradigma verhaftet*; Datenbanken, die gemäss Aufgabenstellung in die dritte Normalform überführt wurden, als *viel zu dogmatisch mit der Normalformen-Scheuklappe modelliert* bewertet. Aufgrund vieler Abwesenheiten des Professors müssen die Studierenden oftmals mit seinem ebenfalls aus dem universitären Fribourger Informatikfilz stammenden Assistenten Denzler vorlieb nehmen, der im Nebenfach scheinbar Anglizistik (sic!) belegt hat (bei Prof. G. Alao?) und auch mit den gängigsten Trump- und Nordkorea-Witzen bestens vertraut ist. But who cares anyway? Zum Ende des Semesters, also zu einer Zeit, an denen sonst kaum etwas zu tun ist, durften sich die einzelnen Gruppen noch an eine Projektarbeit heranwagen, die eine Migration von einer SQL- zu einer NoSQL-Datenbank zum Gegenstand hatte. Die meisten Gruppen begnügten sich dabei mit einer Übungsdatenbank, die sich manuell wesentlich schneller als mit den Mitteln der Informatik (mit sogenannter *Software!*) migrieren lässt.
- DMATH: Dr. Joseph Bürglers praktische Zahlentheorie und theoretische Zahlenpraxis mit theoretischen und praktischen Zählübungen (ohne Fingereinsatz); offiziell als *Diskrete Mathematik* geführt. Das Abzählen von Heeresgrössen nach dem chinesischen Restsatz und die von Euler auf die Königsberger Brücken angewandte Spaziergangstheorie sind nur zwei der in DMATH vermittelten praxisrelevanten Fähigkeiten.

- EENG: English for Engineers (und Wirtschaftsingenieure!) ist ein Modul, das hält, was es verspricht; die bisher einzige Überraschung war das Ausbleiben von Überraschungen. Selbst der Gruppenarbeitsstumpfsinn wird auf ein erträgliches Mindestmass reduziert. Bei der Präsentation, die Teil des Testats ist, sollte man den Begriff «Third World Country» vermeiden. Die politisch korrekte Bezeichnung lautet anscheinend «Shithole Country» (Stand Januar 2018).
- FKOM: *Fachkommunikation* oder Deutsch für Informatiker inklusive einer Einführung in *das Internet* von einem Nicht-Informatiker erteilt an Informatiker. Es wurde gebloggt und einmal fiel sogar das Wort «Schlumpfschach».
- IMATH: In diesem Modul, genannt *Informatik-Mathematik*, erbringt Dr. Joseph Bürgler den endgültigen Beweis dafür, dass man sich Mathematik zwar nicht als Gruppenarbeit aneignen kann, dies aber trotzdem unbedingt versucht werden muss.
- INFKOL: Das Modul *Informatik-Kolloquium* oder *Computer Science Hot Topics* wurde als Energiesparprogramm für Studierende und Dozierende konzipiert und findet am Mittwochabend als Konkurrenzveranstaltung zu verschiedensten ISA-Modulen statt. Das Modul besteht einerseits aus fünf Gastvorträgen, wobei der erste Gastvortrag von einem der Modulverantwortlichen (Rainer Schaub) bestritten – jedoch wie die vier anderen Gastvorträge auch mit einer *Zuger Kirschtorte* vom anderen Modulverantwortlichen (Martin Jud) abgegolten wird. (Böse Zungen behaupten, das Modul INFKOL sei als Lösung für Rainer Schaub's Geburtstagsortenproblem konzipiert worden, zumal er seit der Einführung dieses Moduls seine Tante Elfriede jeden Frühling mit einer *Zuger Kirschtorte* im Gepäck in einem Altersheim im Schwarzwald besucht.) Beim Hochladen der zu jedem Gastvortrag fälligen Zusammenfassung, Fragen und Antworten hat es sich als vorteilhaft erwiesen, wenn der Name der Studierenden jeweils Teil des Dateinamens war, sodass diese Dokumente zur Kontrolle nicht noch zuerst umständlich geöffnet werden mussten. Der zweite Teil des Moduls besteht in einem Term Paper, das von Studierenden alleine oder in Zweiergruppen verfasst wird. Der Begriff des *Computer Science Hot Topic* ist dabei so weit gefasst, dass subjektiv vermutete Unterschiede zwischen Asien und Europa als Thema gelten können.
- IPCV: Unter Anleitung von Thomas Koller, der das Betrachten von Fussballspielen lieber an OpenCV delegiert, lernen die Studenten im Modul *Image Processing and Computer Vision*, wie man das Playmate Lena auf den Kopf stellt, Hände abschneidet, ein X-Wing-Raumschiff auf dem Mond platziert und Einstein scharf macht. Dr. Joseph Bürgler sorgt mit seinem gekonnten Einstreuen von Integral- und Summenzeichen für den angemessenen wissenschaftlichen Touch.
- ISF: Der europareisende Security-Kosmopolit Hämmerli und der altgediente, drittweltgewandte Wirtschaftsinformatik-Haudegen Marfurt wechseln sich im Modul *Information*

- Security Fundamentals* zwischen theoretisch-theoretischen (Hämmerli) und anschaulich-praktischen (Marfurt) Themen im Bereich der Informationssicherheit ab. Assistent Pouly lanciert einen Cyberangriff auf studentische Passwörter, um sie zur Entwicklung ihrer Cyberselbstverteidigungsfähigkeiten anzuregen.
- ITEO: *IT Engineering & Operations* ist eine CNA-Fortsetzung, die auch als *Waldmann-Joho-Kurmannscher Informatik-Almanach* bekannt ist. In diesem Modul kommt nicht nur Power-Point, sondern auch Excel zum Einsatz.
  - MATH: Evaluationsseminar für die E-Learning-Plattform *Learning Catalytics* des Pearson-Verlages anhand verschiedenster Aufgaben zur Differential- und Integralrechnung bei Herrn Dr. Schmidt unter Einsatz eines dem Abakus deutlich überlegenen Taschenrechners; offiziell als *Mathematik-Grundlagen* geführt.
  - KS: Im ISA-Modul *Kreatives Schreiben* and der HSLU – Wirtschaft wird jeden zweiten Mittwochabend – ja! – kreativ geschrieben. Die dazu vom Hausdienst bereitgestellte Kaffeemaschine unterstützt diesen Prozess, wozu ein mit Bier gefüllter Kühlschrank oder eine Flasche Absinthe wohl noch besser geeignet wären.
  - MOD: Ein weiteres Kurmann-Modul, geführt als *Modellieren Grundlagen* mit Schwäbisch-Exkursen erteilt durch den Professor Zimmermann. Die UML-Notation, die bereits im vorhergehenden Semester vorausgesetzt wurde, wird hier nun eingeführt.
  - OOP: Das Modul *Objektorientierte Programmierung*, das gleichzeitig mit PLAB am Montag-nachmittag und gleichzeitig mit dem Auschlafen am Dienstagvormittag stattfand für Studierende, die nicht schon mehrjährig Java-geschädigt sind.
  - MC: Im Modul *Mikrocontroller* erläutert Peter Sollberger (der bereits 1990 den Linux-Kernel selbst kompiliert haben will, obwohl Linus Torvalds die Arbeit daran erst im April 1991 begonnen hatte) interessierten Studenten der Informatik und Elektrotechnik die Programmiersprache C. Dabei wird von der ersten Lektion an eine IDE (NetBeans) verwendet, welche gigantische Boilerplate-Projekte mit entsprechenden `Makefiles` erzeugt, sodass der Dozierende sich eine Viertelstunde lang über den Build-Prozess auslassen kann, währenddem Studenten mit C-Vorwissen einfach `make hello && ./hello` eintippen und anschliessend auf das Ende der Ausführungen warten. Der Einsatz einer schlanken, Linux-basierten Entwicklungsumgebung bestehend aus `vim`, `gcc`, `gdb` und `make` wird dabei geächtet, sowie das eigenhändige Schreiben einfachster `Makefiles`. Stattdessen ist unbedingt eine IDE wie NetBeans zu verwenden, denn nur damit lasse sich professionell C programmieren. (Würden doch Linus Torvalds, Ken Thompson und Rob Pike nur auf Professor Sollberger hören!) Im zweiten Teil des Moduls wird unter Anleitung des Dozierenden Christian Jost die Installation weiterer Bloatware eingeübt, womit ein Mikrocontroller mit äusserst knappen Ressourcen programmiert

werden soll.

- PCP: Im Modul *Programming Concepts and Paradigms* geht es einerseits bei Ruedi Arnold um die deklarativ-logische Programmierung in Prolog mit SWI-Prolog, die im starken Kontrast zur spekulativ-unlogischen Programmierung in Java steht, wie sie sonst an der HSLU betrieben wird; andererseits bei Roger Diehl um die deklarativ-funktionale Programmierung in Scheme mit Racket, wobei lexikalisches Scoping (was schon alle kennen) und prozedurale Programmierung (was man damit nicht machen sollte) behandelt werden (was infolgedessen niemanden interessierte). Die Behandlung des funktionalen Paradigmas beschränkt sich auf die Übergabe von Funktionsparametern und trivialste Lambda-Expressions. Makros (das Alleinstellungsmerkmal der Sprache) und Closures werden nicht erwähnt, geschweige denn behandelt. Funktionale Programmierung in Java 8 (d.h. `filter`, `map`, `reduce` auf Streams) wird hingegen an vier Vormittagen mit Foliensätzen à 50 MB ausgiebig behandelt. Die Aufgaben müssen regelmässig erledigt und den Dozenten vorgezeigt werden, wobei die zu lösenden Probleme unter anderem von Ruedi Arnolds *Problem Provider* generiert werden, bei dem der Name Programm und die zugrundeliegende Programmiersprache naturgemäss Java ist. («I had a problem and thought to use Java, now I have a ProblemFactory.») Die grösste Schwierigkeit neben dem JSON-Handling in Prolog ist das schnelle Eintragen auf das Aufgaben-Kontroll-Doodle, denn nur wer sich früh genug in die Liste einträgt und die Aufgaben bald vorzeigen darf, kann auch das Mittagessen bereits um 11:20 Uhr einnehmen und sich so frühzeitig auf das anstehende Rentnerdasein einstellen.
- PLAB: Die bereits in Java bewanderten Studierenden durften unter der Anleitung von Prof. Ruedi Arnold und dem externen Aushilfsassistenten Joachim (ohne Nachnamen) ihre bereits vorhandenen Java-Kenntnisse durch selbstgehaltene Vorträge und Kurzübungen im Modul *Programmierlabor* vertiefen.
- PMB: Projektmanagement, theoretische Physik und *Mensch & Umwelt* in einem. Es wird ein fiktives Konzept für die fiktive (und zum Untergang verdamnte) Firma SwissGrooves (die sogenannte «Rille») ausgearbeitet, das den Studierenden wohl die Unbedingtheit des Scheiterns (und dessen Erscheinung im Raum-Zeit-Kontinuum (in seiner physisch-reellen Materialisierung (in der seienden Seinswelt (veranschaulicht durch eine hermeneutischer Dialektik (anhand eines teleologisch-analytischen Ansatzes (usw. usf.))))))) vermittelt durch Herrn Prof. Dr. nat. phil. Martin Jud im Rahmen des Moduls *Project Management Basics* veranschaulichen soll.
- PMRE: Aufgrund spärlicher Anwesenheit im Modul *Project Management und Requirements Engineering* kann der Autor dieser Zeilen keine qualifizierten Aussagen darüber machen und muss über den Umweg der Spekulation ausgehend von Modulunterlagen («das System soll

auf der Basis <WWW-Browsertechnologie> entwickelt werden») zu seinem Urteil gelangen, das aufgrund seiner Voreingenommenheit gegen Wirtschaftsinformatiker hiermit «grober Unfug» lauten soll.

- PREN I: Im Flaggschiff unter den Projektmodulen *Produktentwicklung 1* ging es im Herbstsemester 2017 um die Konzipierung einer autonomen sogenannten *Laufkatze*, die eine Last orten, aufnehmen und punktgenau im Zielbereich ablegen soll. Die Gruppengrösse übertrifft mit sieben bis acht jedes vernünftige Mass, die Organisation in Horw ist aber derjenigen in Rotkreuz um Lichtjahre voraus. Oft wird auf das Modul *Kontext 1* verwiesen, das bei den Rotkreuzern unter den Studierenden nur als Verweis in ein geistiges Vakuum verstanden wird. Melancholisch veranlagte Studierende geraten dabei schon einmal ins Grübeln («Ach dort hätten wir etwas über *die gute Projektführung* erfahren sollen!»)
- PREN II: Im Frühlingsemester 2018 wurde umgesetzt, was im Herbstsemester zuvor konzipiert wurde. Das Modul *Produktentwicklung 2* endet mit einem Wettkampf, der für viele Dozenten sichtlich und trotz anstehender Semesterferien den existenziellen Höhepunkt des jeweiligen Jahres darstellt. Die Notengebung erfolgt mit einem speziellen Würfel, dessen Wurfresultate normalverteilt sind. Erfolgreich umgesetzte und mit «*sehr gut*» bewertete Konzepte werden so etwa als «*ausreichend*» bewertet. Funktionstüchtigkeit des Prototypen und Schlussbewertung weisen keine Korrelation auf.
- PTA: Mit der sogenannten *Projekt- und Teamarbeit* (auf Englisch: *Pointless Team Assignment*) wurde der absolute Tiefpunkt des Studiums strategisch an den Anfang desselben gesetzt. Das gegenseitige Kennenlernen in Sechsergruppen – gemäss Modulverantwortlichem Hofstetter für dieses Modul essenzieller Bestandteil und dessen eigentlicher Zweck – fand aufgrund missverständlicher Planung leider erst gegen Ende der ersten Semesterhälfte statt. Die Reflexionen, bei denen es nur scheinbar um die Reflexion der zu erstellenden Arbeit zum Thema «Digitale Mobilität» ging, diente den Studierenden dazu, sich Gedanken darüber zu machen, was sie denn an dieser Institution überhaupt verloren haben. Besonders motivierend waren dabei die vernichtenden Rückmeldungen auf unbefriedigend präsentierte Zwischenergebnisse von Herr Dr. Adrian Aebi. Die Auslosung des Präsentierenden bei der Abschlusspräsentation, die den grössten Teil der Gesamtbewertung ausmachte, sollte den Studierenden wohl den hohen Stellenwert der Variable Zufall für den weiteren Studienverlauf und künftigen Lebensweg vor Augen führen und sie dadurch zur demütigen Einkehr anhalten. Die Welt ist ungerecht – die HSLU *grausam*. Die eigentliche Projektarbeit wurde durch verschiedene sogenannte *Inputs* ergänzt:
  - Präsentieren I-XXVI: Videoanalyse studentischer Kurzpräsentationen mit packendem Livekommentar vom Dozent Daniel Albert.

- Literaturrecherche: Einführung in die Literaturrecherche erteilt am Einführungstag, sodass der Stoff bis zur Anwendung garantiert wieder vergessen wurde.
- Design Thinking: 30 Studierende sitzen sich in einem Raum gegenüber und basteln zu House-Musik eine Brieftasche aus Moosgummi, Schnur und Papier bis einer der Dozenten Gugolz oder Zwimpfer das Ende der Übungsphase mit einer Hupe signalisierten. (Gleich zwei Professoren mit Batikhemden können nichts Gutes verheissen!)
- SELL: Das aufgrund seiner geringen Pflichtenwahrnehmung äusserst beliebte *Self Directed English Language Learning*, bei dem in Zweiergruppen über ein selbst gewähltes Thema referiert werden musste, ist das ideale Alternativprogramm zum mittwöchlichen Feierabendbier.
- SPRG: Im Modul *Sicheres Programmieren* haucht Bernhard Hämmerli (in Indien auch bekannt als Prof. Bernhard Hamarily/Hamarilli von der Lucini University) der K.u.K-Monarchie neues Leben ein, indem er zwischen einer österreichischen und ungarischen Besetzung variiert: Einmal der von seinen norwegischen Wanderferien bekannte Herr Prof. Dr. Paul Tavolato aus St. Pölten; einmal der von ihm sogenannte *hōngärian Prōgrämmer*, den er mutmasslich auf dem Weg zu einem Altersheimbesuch («*to visit my möther*») kennengelernt hat. Das Modul findet (der Modulausschreibung entgegen) teilweise ganztagig am Samstag statt, aber wer damit ein Problem habe, könne sich ja gerne wieder abmelden, meint das HSLU-Sekretariat dazu. Die letzte Modulveranstaltung fällt dabei nicht wie im Modul SWT aus familiären, sondern aus bis dato unbekannten Gründen aus. (Für eine Stellungnahme waren weder der Modulverantwortliche Hämmerli noch der «*hōngärian Prōgrämmer*» erreichbar.) Die Modulunterlagen werden teilweise (für die Studenten nicht einsehbar) auf der Übungsablage, teilweise (durch die Studenten modifizier- und löschbar) auf dem Dateiaustausch abgegeben, sodass nur die regelmässigen Archivare unter den Studierenden überhaupt beweisen konnten, dass die letzte Modulveranstaltung überhaupt im Semesterplan erwähnt worden war.
- SWT: Im Modul *Software Testing* stellt der Dozent Dominique Portmann die Studenten vor unlösbare Aufgaben, indem er diese vor unlösbare Aufgaben stellt, um zu demonstrieren, dass sich unlösbare Aufgaben nicht lösen lassen, da diese nicht lösbar seien, und schon gar nicht mit *Selenium*. Wichtige IT-Fragestellungen, etwa wie man Tee zubereitet oder wie man einen Ball testet, sind eine willkommene Abwechslung zu den mehrstündigen PowerPoint-Karaoke-Sessions, die teilweise mit eigenen und teilweise mit fremden Folien bestritten werden. Für die Testatprüfung kommt Excel (oder war es «Exel»?) zum Einsatz, wobei das Worksheet mit einer ausgeklügelten Kombination von einem Testing-Begriff und einer SCB-Schreibweise passwortgeschützt ist. Die letzte Modulveranstaltung kann dabei schon einmal aus «familiären Gründen» ausfallen, gerade wenn diese auf einen Freitagabend nach dem Auffahrtsdon-

nerstag fällt. Als Alternativprogramm zur PowerPoint-Karaoke beim SCB-Fan Portmann an der HSLU bietet sich am jeweiligen Freitag das Blick-Playback bei den EVZ-Fans in der örtlichen avec-Filiale an, das kognitiv, rhetorisch und logisch auf einem ähnlich hohen Niveau angesiedelt ist und mit einer vergleichbaren Inbrunst und Wehemenz vorgetragen wird. Ersetze man den Ausruf «*verdammti huere Schoofseckle!*» mental durch «*Teststrategie!*», ist das intellektuelle Flair einer SWT-Vorlesung auch in der avec-Filiale zu verspüren, wobei der Lernprozess in letzterem Setting durch verschiedene Sorten Dosenbier unterstützt wird.

- VSK: Im sagenumwobenen Modul *Verteilte Systeme und Komponenten* geht es darum, eine von der HSLU unter HSLU-Standards als *schlecht* beurteilte Software um eine austauschbare, client-seitige Logger-Komponente zu erweitern. Diese Komponente ist insofern austauschbar, dass bei der Verwendung einer fremden Logger-Komponente auch der ganze dazugehöriger Server-Stack ausgetauscht werden muss. Um die Studierenden auf die anstehende Projektarbeit vorzubereiten doziert Herr Prof. Dr. nat. phil. Martin Jud unter anderem zu den Themen «*Die Struktur des Strukturlosen: ein Kriegsbericht von der Modulplanungsfront*»; «*Chaos als ästhetische Kategorie oder: wie man Java-Packages strukturiert*» und schliesslich «*Irrungen, Wirrungen: was uns die späte Prosa Fontanes über die Konzeption und Planung von Software nach dem SoDa-Vorgehensmodell zu sagen hat*». Nach der Vorlesung zum Thema «*Die Projektinitialisierung: Im Spannungsfeld zwischen Erwartungsangst und Erscheinungsschrecken*» konnten sich die Gruppen an die Projektarbeit machen, wobei Studierende jüngerer Jahrgänge zum ersten – und letzten! – mal in ihrem Leben ein Java-Applet ausführten. Der Versuchung zu widerstehen, ungeliebte Tasks in ScrumDo Herrn Hofstetter zuzuweisen, stellt die Hauptschwierigkeit in der Projektarbeit dar. Innovation war bei der Prüfungsvorbereitung gefragt, denn an der MEP musste eine Tabelle mit mehreren Reviewarten und zahlreichen dazugehörigen Aspekten hergeleitet werden – und zwar 1:1 aus dem Gedächtnis. Die Studierenden lösten das Problem schliesslich mit einem rhythmischen Klatsch- und Patsch-Spiel. Neben theoretischem Wissen werden auch zahlreiche praktische Prinzipien eingeführt:

- Open Closed Principle (OCP): Man ist für Veränderungen offen, das Sekretariat für deren Umsetzung aber bereits geschlossen.
- Single Responsibility Principle (SRP): Roland Gisler ist alleine für die Modulqualität verantwortlich.
- Separation of Concerns (SOC): Martin Jud erarbeitet die Prüfung, der Computer korrigiert sie.
- Principle of Least Surprise (PLS): Positive Überraschungen bleiben wie immer aus.
- Don't Repeat Yourself (DRY): Auch mehrmaliges Nachfragen führt nicht zu einer verständlichen Erklärung.



- Keep It Simple, Stupid (KISS): Bei der MEP gibt es volle Punktzahl – oder gar keine Punkte.
- WEBTEC: Beim Modul *Web Technologies* handelt es sich um Peter Infangers *tour de force* durch die Web-Technologien HTML, CSS und JavaScript etc. mit Zurschaustellung seiner privaten Büchersammlung zu selbigen Themen und Surftipps wie [rotten.com](http://rotten.com) ist ein beliebtes und kurzweiliges Donnerstagnachmittagsvergnügen, das unter Studierenden auch unter der Modulkategorie «betreutes Surfen» geführt wird. Mit einem 6.9 Gigabyte grossem Ubuntu-Image für ein *Hello World*-Beispiel mit AJAX kommt dabei *cutting edge*-Technologie höchster Güte zum Einsatz. Bei der Modulendprüfung ist ein hohes Bearbeitungstempo wichtig, damit man möglichst bald wieder gehen kann.