Developing Software Testing Courses for Your Staff

Cem Kaner, J.D., Ph.D.

Workshop at the Pacific Northwest Software Quality Conference

October 9, 2006

These notes are partially based on research that was supported by NSF Grant EIA-0113539 ITR/SY+PE: "Improving the Education of Software Testers." Any opinions, findings and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation.

Meet & Greet

- *Who are you?*
- *What are the most important three things we should know about your background?*
- *Why are you here?*
- *What do you hope to leave with?*

Source Materials on the Disk

- *Video lectures*
- *Activities*
- *Assignments*
- *Sample exam questions*
- *Some readings*
- *A little additional instructional support material*

- *These materials echo sites that James Bach, Scott Barber, Tim Coulter, Rebecca Fiedler and I have been creating at Florida Tech (www.testingeducation.org) and Satisfice (www.satisfice.com) that will give access to reusable content and host supervised courses. (Some of the Satisfice-site courses will cost money, others will be free.)*
- *All of my instructional materials are available, royalty-free, under the Creative Commons license.*

The Underlying Problem

- *Most of today's software testing techniques were developed in the 1970's.*
  - *Back then, long programs were 10,000 statements*
  - *Code was often readable COBOL*
  - *An enterprising tester could read the entire program, identify all the variables and most of the relevant combinations.*

The Underlying Problem

- *Over the past few decades, programmer productivity has surged, driven by revolutions in software engineering practice.*
- *Class libraries make it easy to snap together large applications*
- *Many consumer products include millions of lines of code.*

The Underlying Problem

- *We have not experienced revolutions in testing practice and we are not much more productive today than we were three decades ago:*
  - *Regression test automation offers small, incremental improvements in productivity*
  - *High-volume test automation is still rarely done and is poorly understood by the general (e.g. academic) testing community*
  - *Test case documentation is as overblown as ever, with a new generation of semi-automated "test-case management" bureaucracy to slow us down further.*

The Underlying Problem

- *We are much better at*
  - *testing*
  - *documenting the testing*
  - *reporting status of the testing*

  of a 10,000 statement program.

- *But as the size of programs grows geometrically*
- *and the efficiency of testers grows maybe linearly*
- *we impact less of the program every year.*

- *System-level testing will become irrelevant, because we will impact so little of the product.*

Alternatives to Consider

- *Maybe black-box, system-level testing is obsolete*
- *Maybe certifications will assure skill in our field*
- *Maybe university training in testing will foster skill and the evolution and spread of new paradigms*

Maybe we have to revolutionize commercial training

Maybe Black-Box System-Level Testing is Obsolete

- *It is wildly inefficient to expose unit-level bugs (like input-field filter bugs) with system-level tests.*
- *However, many aspects of software behavior emerge in the broader system, not at the unit level:*
  - *Race conditions*
  - *Stack corruption, memory leaks, odd error handling*
  - *Utility, security, performance, etc.*

Maybe Black-Box System-Level Testing is Obsolete

- *If*

Quality is value to some person

--Weinberg

- *Then we still need to investigate the extent to which the product under development provides or fails to provide appropriate value to the relevant stakeholders.*
- *Unit testing doesn't address this*
- *Simplistic functional testing, including "story" testing, barely begins to address this.*

Maybe Black-Box System-Level Testing is Obsolete

The need for this type of investigation is still present.

Whether we can competently satisfy that need remains to be seen.

Maybe Certification Will Assure Skill

- *Current certification exams focus on superficial knowledge (e.g. definitions, memorizable descriptions, etc.) rather than evaluation of skilled performance*
- *Review courses that teach you how to pass the certification exam are very commercially successful at the moment, but I question their educational value (more later in these slides).*
- *The "bodies of knowledge" that I've reviewed seem firmly grounded in 1970's/1980's material.*
- *I don't see how this moves us forward*

Maybe university training in testing will foster skill and evolution / spread of new paradigms

- *Universities have played a large role as change agents in other fields (including programming & design)*
  - *New development paradigms turn into new courses or rewrite old ones:*
    - *UML is mainstream*
    - *OO design is mainstream*
    - *Test-first programming is spreading in early courses*
  - *New graduates infiltrate new approaches into traditional workgroups*

Maybe university training in testing will foster skill and evolution / spread of new paradigms

- *We considered a testing degree at Florida Tech*
  - *Rich enough intellectual problem to deserve a degree.*
  - *But serious risk of career stereotyping and lock-in*
  - *Abandoned in favor of a software engineering degree that offers*
    - *Courses in black box software testing and programmer testing*
    - *additional test-relevant courses (e.g. human factors) and testing options.*
  - *This is better than most other places but still just scratches the surface.*

Maybe we have to deal with it as industrial training

- *Industrial training has its own challenges*
- *I left a very successful consulting practice*
  - *whose main income generator was commercial training*
  - *accepting a 2/3 reduction in income*
  - *because I concluded that most commercial training is good for introducing new concepts but not good for developing skills or critical insight.*
- *If we're going to foster deeper skill development and richer evaluation of practice – if we want to kickstart the next productivity revolution in testing – the short-course model won't do it.*

Commercial vs. Academic

• *Drive-by teaching* – *2-5 days, rapid-fire ideas, visiting instructor*	• *Local teaching* – *Several months, a few hours per week, students get to know instructor*
• *Broad, shallow coverage*	• *Deeper coverage*
• *Time constraints limit activities*	• *Activities expected to develop skills*
• *No time for homework*	• *Extensive homework*
• *No exams*	• *Assessment expected*
• *Coached, repeated practice seen as time-wasting*	• *Coached, repeated practice is highly appreciated*
• *Familiarity*	• *Capability*
• *Work experience helps to bring home concepts*	• *Students have no work experience, need context*
• *Richer grounding in real practice*	• *Harder to connect to real practice*
• *Some (occasional) student groups share a genuine current need*	• *Students don't naturally come to a course as a group with a shared problem*
• *Objective: one applicable new idea per day*	• *Expect mastery of several concepts and skills*

My idea has been to develop courses in an academic environment (where I can learn more about what works and why), with the goal of providing an alternative model for commercial (in-house) training and professional self-study

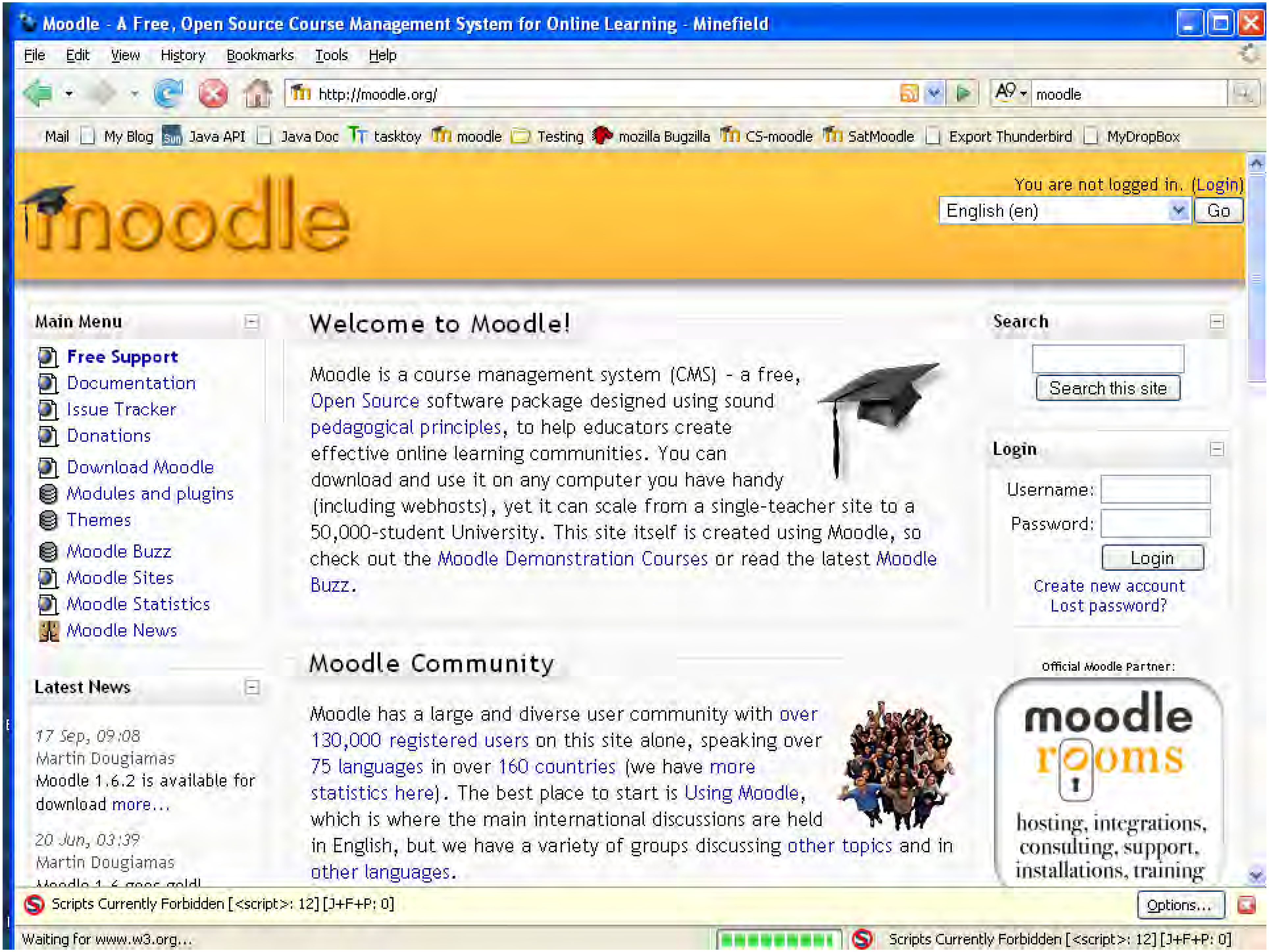
Today's workshop is a progress report against a broader curricular vision

Overview

1. *Tour of the Moodle course management system*
2. *Tour of the Black Box Software Testing Course on Moodle*
3. *Overview of use of material like this in the workplace*
4. *Dealing with the instructional challenges of a cognitively complex field of study*
5. *Taking control of your learning objectives for the course*
6. *Examples of activity patterns*

An Overview of Moodle

- *www.moodle.org*
- *Free*
- *Course management system*
- *Useful for:*
  - *Live short courses (requires web access)*
  - *Live academic courses (long term, homework)*
  - *Hybrid of remote / live*
  - *Remote courses*
    - *Synchronous (live web conference tools are better)*
    - *Asynchronous*



Welcome to Moodle!

Moodle is a course management system (CMS) - a free, Open Source software package designed using sound pedagogical principles, to help educators create effective online learning communities. You can download and use it on any computer you have handy (including webhosts), yet it can scale from a single-teacher site to a 50,000-student University. This site itself is created using Moodle, so check out the Moodle Demonstration Courses or read the latest Moodle Buzz.

Moodle Community

Moodle has a large and diverse user community with over 130,000 registered users on this site alone, speaking over 75 languages in over 160 countries (we have more statistics here). The best place to start is Using Moodle, which is where the main international discussions are held in English, but we have a variety of groups discussing other topics and in other languages.

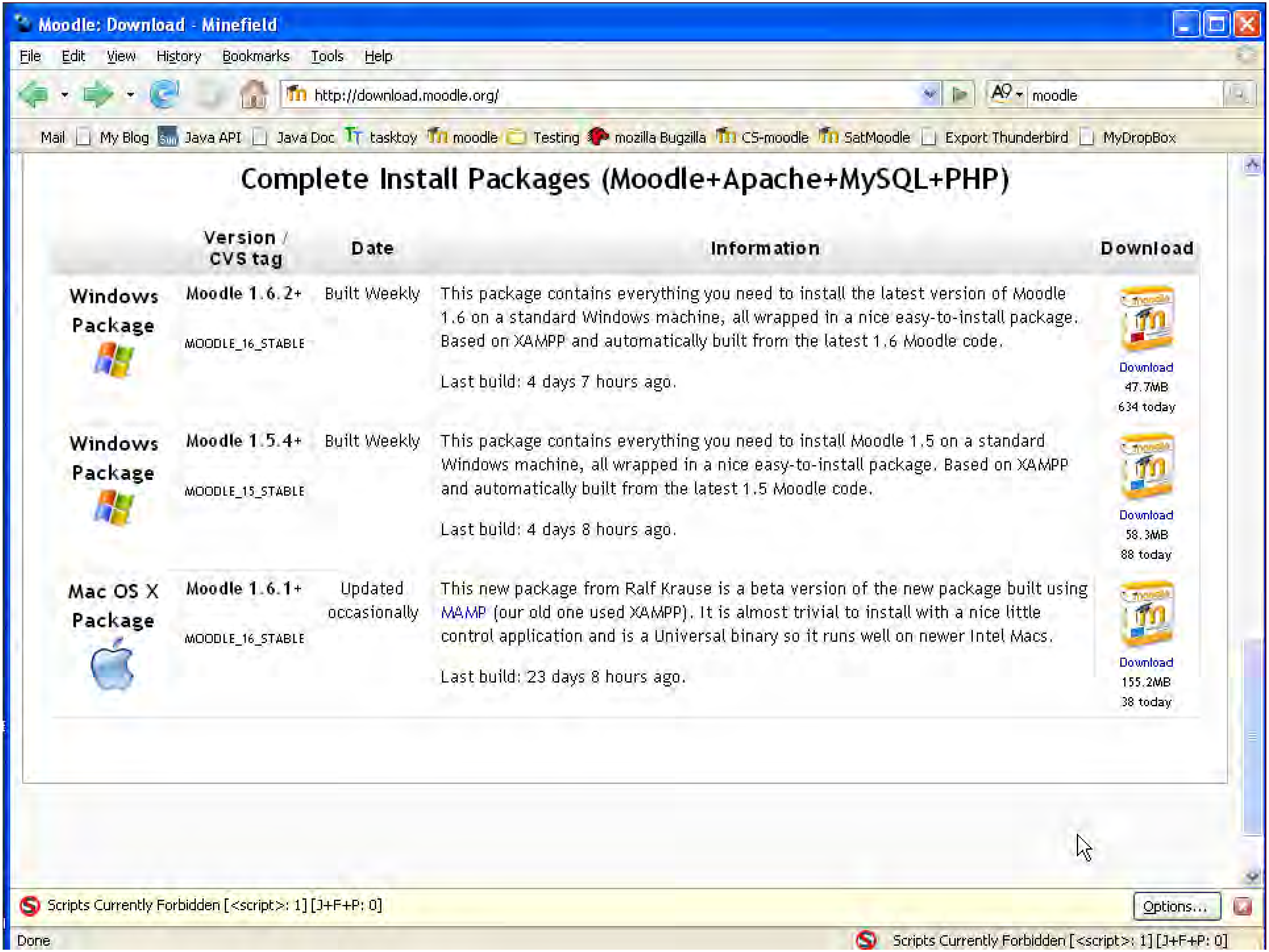
17 Sep, 09:08
Martin Dougiamas
Moodle 1.6.2 is available for download more...

20 Jun, 03:39
Martin Dougiamas
Moodle 1.6 goes gold!

Scripts Currently Forbidden [<script>: 12] [J+F+P: 0]

Moodle platforms

- *Windows*
- *Mac*
- *Linux*



Complete Install Packages (Moodle+Apache+MySQL+PHP)

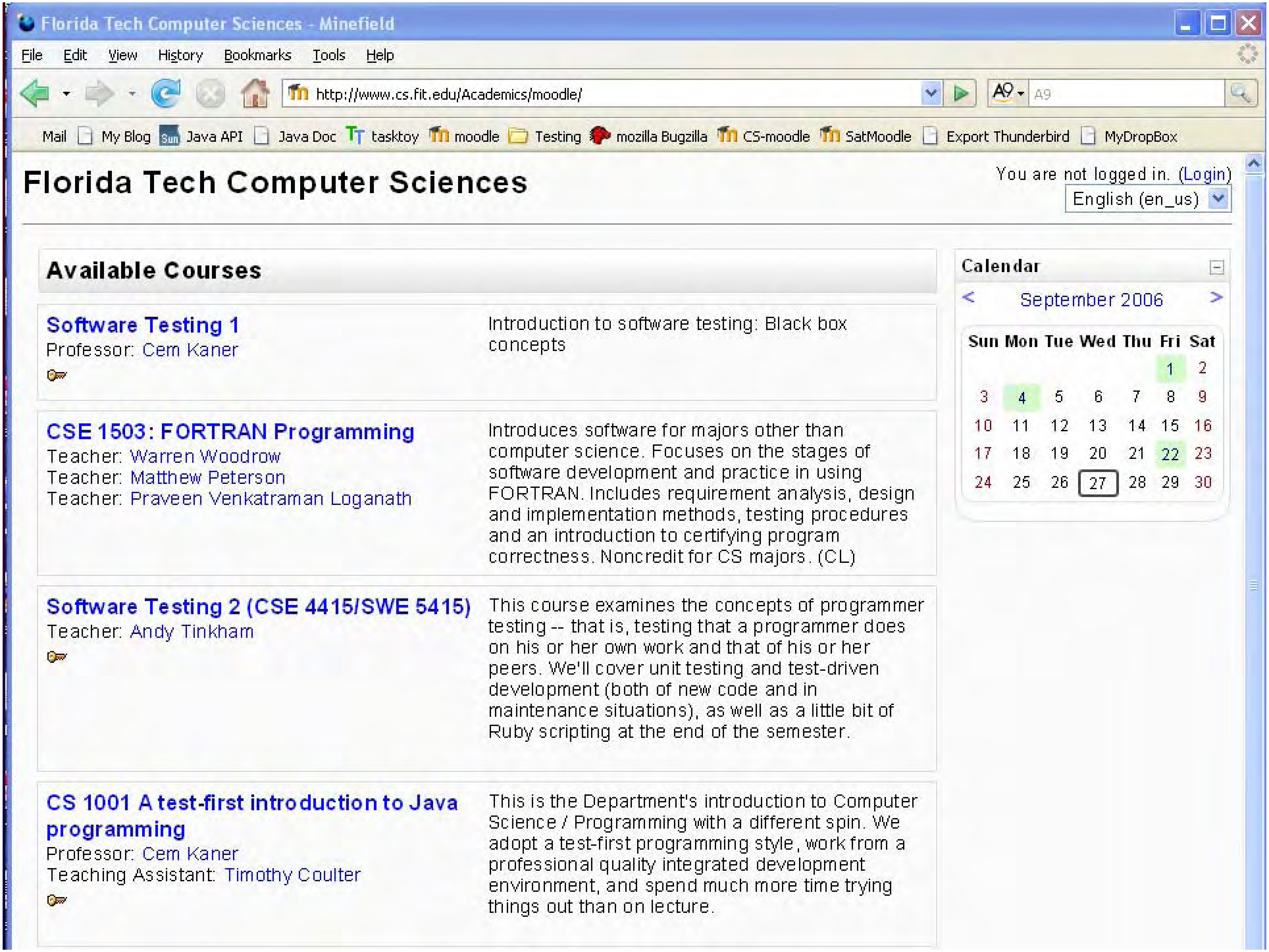
	Version / CVS tag	Date	Information	Download
Windows Package	**Moodle 1.6.2+** MOODLE_16_STABLE	Built Weekly	This package contains everything you need to install the latest version of Moodle 1.6 on a standard Windows machine, all wrapped in a nice easy-to-install package. Based on XAMPP and automatically built from the latest 1.6 Moodle code. Last build: 4 days 7 hours ago.	Download 47.7MB 634 today
Windows Package	**Moodle 1.5.4+** MOODLE_15_STABLE	Built Weekly	This package contains everything you need to install Moodle 1.5 on a standard Windows machine, all wrapped in a nice easy-to-install package. Based on XAMPP and automatically built from the latest 1.5 Moodle code. Last build: 4 days 8 hours ago.	Download 58.3MB 88 today
Mac OS X Package	**Moodle 1.6.1+** MOODLE_16_STABLE	Updated occasionally	This new package from Ralf Krause is a beta version of the new package built using MAMP (our old one used XAMPP). It is almost trivial to install with a nice little control application and is a Universal binary so it runs well on newer Intel Macs. Last build: 23 days 8 hours ago.	Download 155.2MB 38 today

Scripts Currently Forbidden [<script>: 1] [J+F+P: 0]

Scripts Currently Forbidden [<script>: 1] [J+F+P: 0]

Overview of Moodle

- *The following are samples from some courses / activities that I host on moodle*
- *Some data / demonstrations are unavailable (e.g. layout of quiz results) because of student confidentiality rules*



Florida Tech Computer Sciences

You are not logged in. (Login)

English (en_us)

Available Courses

Software Testing 1
Professor: Cem Kaner

Introduction to software testing: Black box concepts

CSE 1503: FORTRAN Programming
Teacher: Warren Woodrow
Teacher: Matthew Peterson
Teacher: Praveen Venkatraman Loganath

Introduces software for majors other than computer science. Focuses on the stages of software development and practice in using FORTRAN. Includes requirement analysis, design and implementation methods, testing procedures and an introduction to certifying program correctness. Noncredit for CS majors. (CL)

Software Testing 2 (CSE 4415/SWE 5415)
Teacher: Andy Tinkham

This course examines the concepts of programmer testing -- that is, testing that a programmer does on his or her own work and that of his or her peers. We'll cover unit testing and test-driven development (both of new code and in maintenance situations), as well as a little bit of Ruby scripting at the end of the semester.

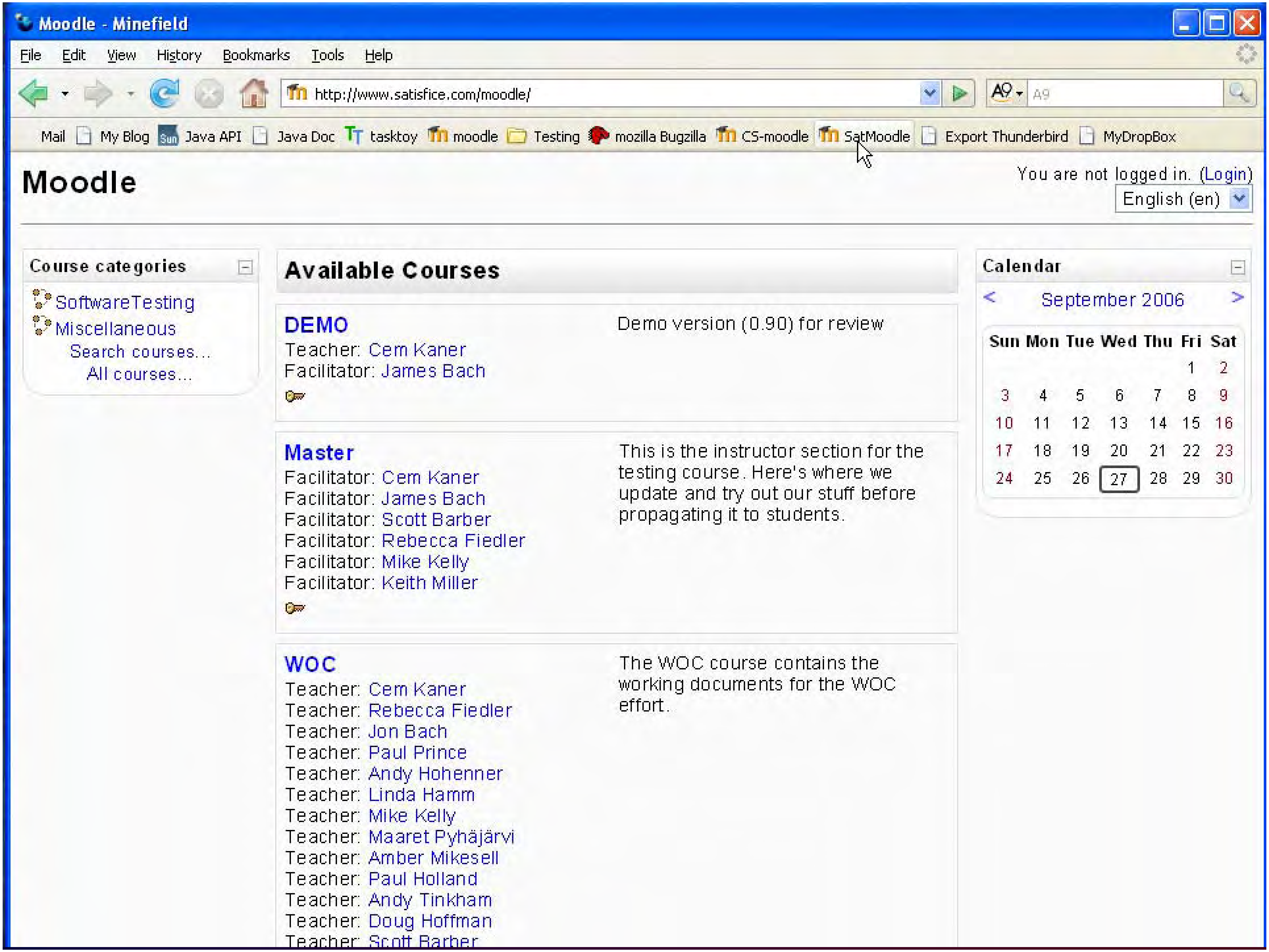
CS 1001 A test-first introduction to Java programming
Professor: Cem Kaner
Teaching Assistant: Timothy Coulter

This is the Department's introduction to Computer Science / Programming with a different spin. We adopt a test-first programming style, work from a professional quality integrated development environment, and spend much more time trying things out than on lecture.

Calendar

< September 2006 >

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30



Moodle

You are not logged in. (Login)

English (en)

Course categories

- SoftwareTesting
- Miscellaneous

Search courses...

All courses...

Available Courses

DEMO

Teacher: Cem Kaner
Facilitator: James Bach

Demo version (0.90) for review

Master

Facilitator: Cem Kaner
Facilitator: James Bach
Facilitator: Scott Barber
Facilitator: Rebecca Fiedler
Facilitator: Mike Kelly
Facilitator: Keith Miller

This is the instructor section for the testing course. Here's where we update and try out our stuff before propagating it to students.

WOC

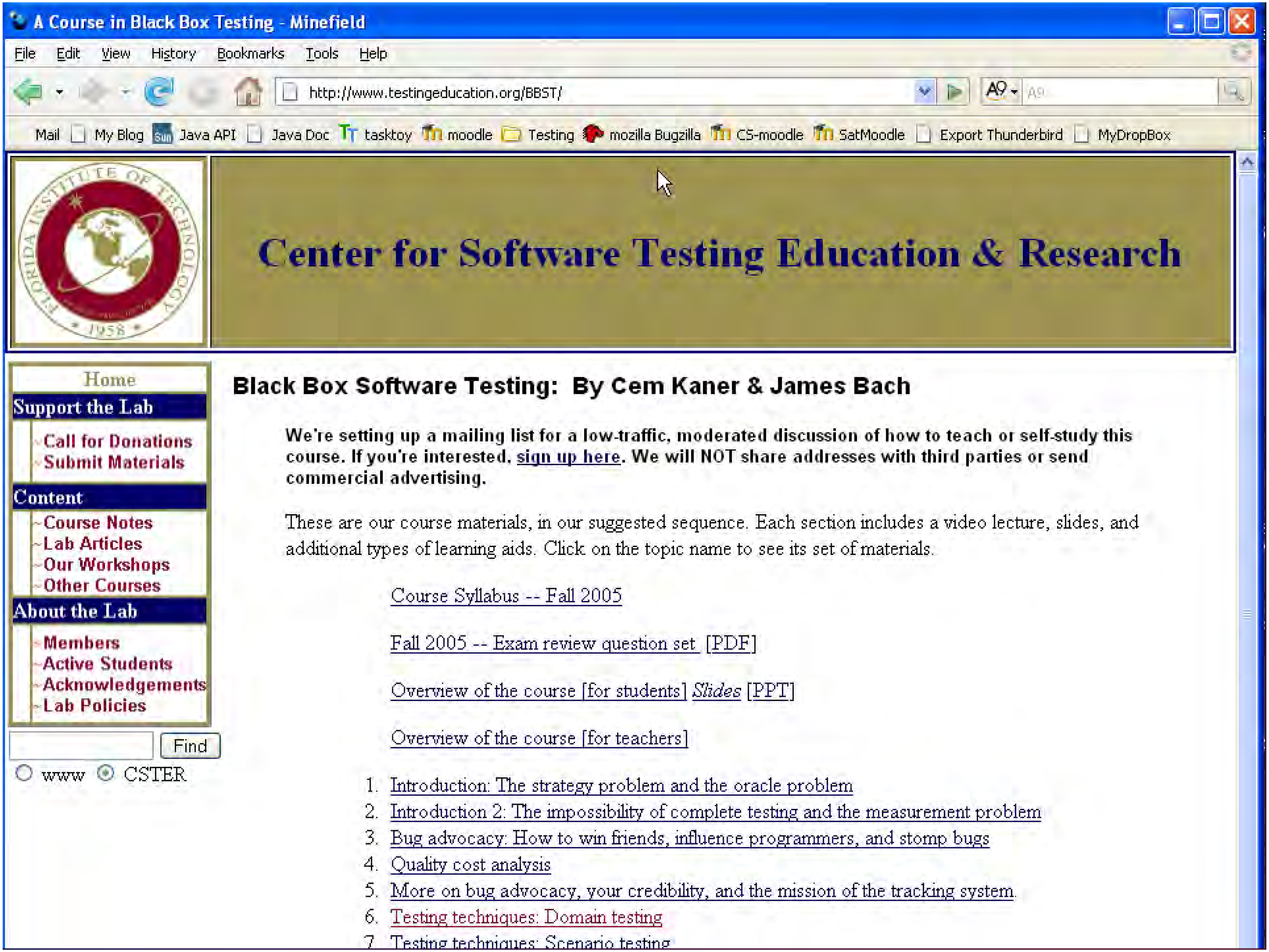
Teacher: Cem Kaner
Teacher: Rebecca Fiedler
Teacher: Jon Bach
Teacher: Paul Prince
Teacher: Andy Hohenner
Teacher: Linda Hamm
Teacher: Mike Kelly
Teacher: Maaret Pyhäjärvi
Teacher: Amber Mikesell
Teacher: Paul Holland
Teacher: Andy Tinkham
Teacher: Doug Hoffman
Teacher: Scott Barber

The WOC course contains the working documents for the WOC effort.

Calendar

< September 2006 >

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30


A Course in Black Box Testing - Minefield
File Edit View History Bookmarks Tools Help
http://www.testingeducation.org/BBST/
Mail My Blog Java API Java Doc tasktoy moodle Testing mozilla Bugzilla CS-moodle SatMoodle Export Thunderbird MyDropBox
FLORIDA INSTITUTE OF TECHNOLOGY 1958
Home
Support the Lab
Call for Donations
Submit Materials
Content
Course Notes
Lab Articles
Our Workshops
Other Courses
About the Lab
Members
Active Students
Acknowledgements
Lab Policies
Find
○ www ● CSTER

Center for Software Testing Education & Research

Black Box Software Testing: By Cem Kaner & James Bach

We're setting up a mailing list for a low-traffic, moderated discussion of how to teach or self-study this course. If you're interested, sign up here. We will NOT share addresses with third parties or send commercial advertising.

These are our course materials, in our suggested sequence. Each section includes a video lecture, slides, and additional types of learning aids. Click on the topic name to see its set of materials.

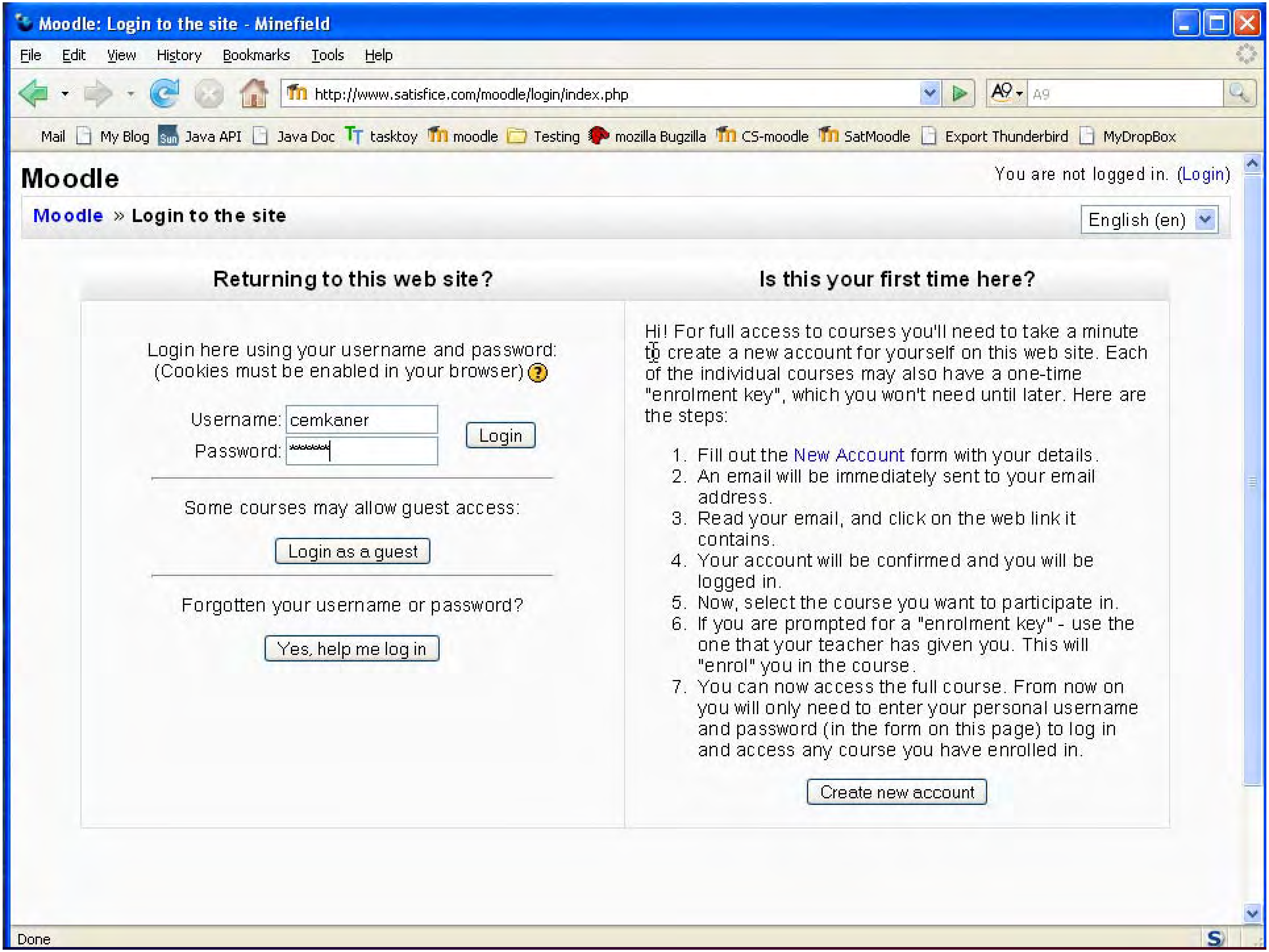
Course Syllabus -- Fall 2005

Fall 2005 -- Exam review question set [PDF]

Overview of the course [for students] *Slides* [PPT]

Overview of the course [for teachers]

1. Introduction: The strategy problem and the oracle problem
2. Introduction 2: The impossibility of complete testing and the measurement problem
3. Bug advocacy: How to win friends, influence programmers, and stomp bugs
4. Quality cost analysis
5. More on bug advocacy, your credibility, and the mission of the tracking system.
6. Testing techniques: Domain testing
7. Testing techniques: Scenario testing



http://www.satisfice.com/moodle/login/index.php

Moodle

You are not logged in. (Login)

Moodle » Login to the site

Returning to this web site?

Login here using your username and password:
(Cookies must be enabled in your browser)

Username: cemkaner
Password: *******

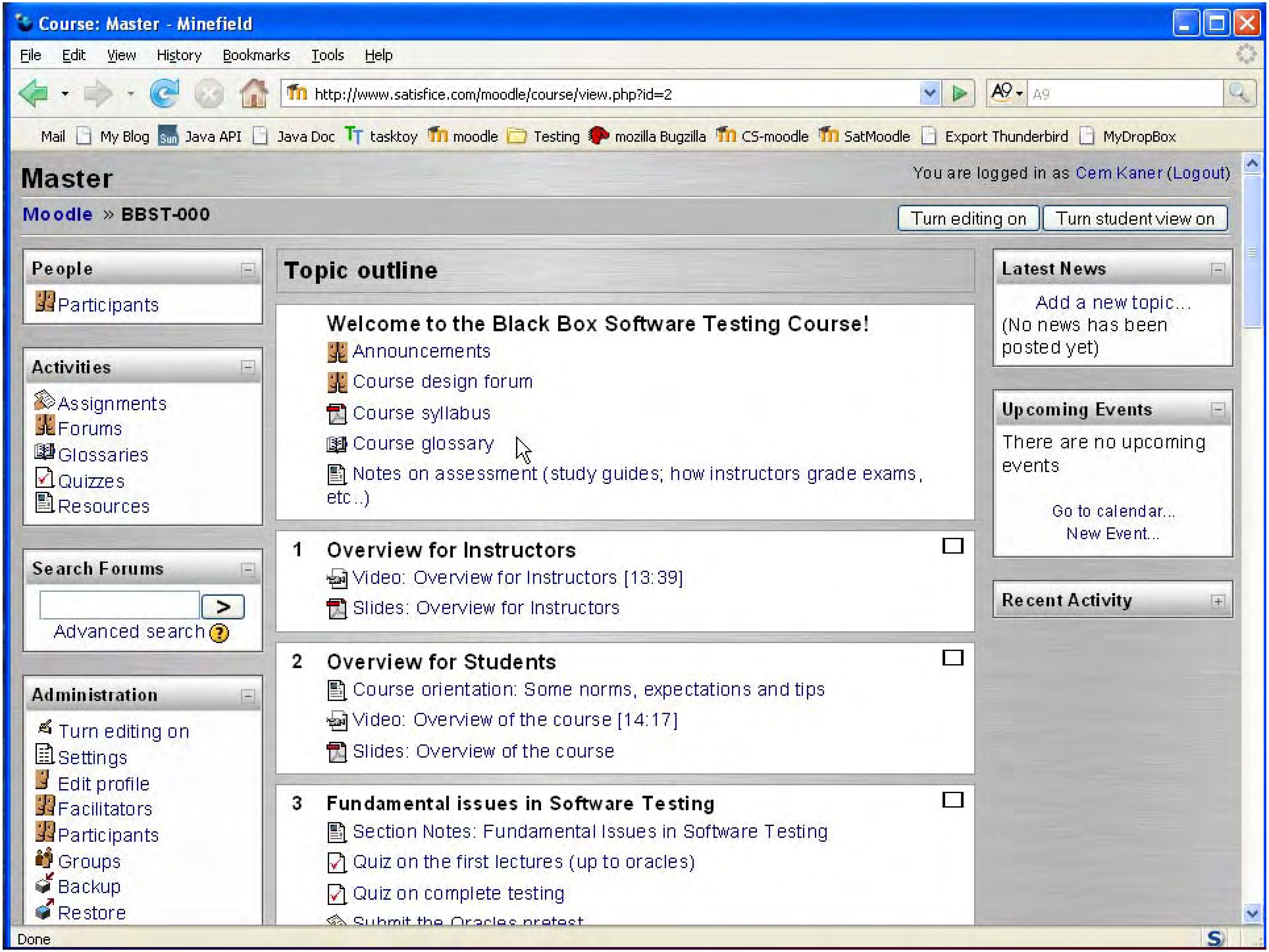
Some courses may allow guest access:

Forgotten your username or password?

Is this your first time here?

Hi! For full access to courses you'll need to take a minute to create a new account for yourself on this web site. Each of the individual courses may also have a one-time "enrolment key", which you won't need until later. Here are the steps:

1. Fill out the New Account form with your details.
2. An email will be immediately sent to your email address.
3. Read your email, and click on the web link it contains.
4. Your account will be confirmed and you will be logged in.
5. Now, select the course you want to participate in.
6. If you are prompted for a "enrolment key" - use the one that your teacher has given you. This will "enrol" you in the course.
7. You can now access the full course. From now on you will only need to enter your personal username and password (in the form on this page) to log in and access any course you have enrolled in.



Course: Master - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/course/view.php?id=2
A9
Mail My Blog Java API Java Doc tasktoy moodle Testing mozilla Bugzilla CS-moodle SatMoodle Export Thunderbird MyDropBox
Turn editing on
Turn student view on
Done

Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000

People

Participants

Activities

Assignments
Forums
Glossaries
Quizzes
Resources

Search Forums

Advanced search

Administration

Turn editing on
Settings
Edit profile
Facilitators
Participants
Groups
Backup
Restore

Topic outline

Welcome to the Black Box Software Testing Course!

Announcements
Course design forum
Course syllabus
Course glossary
Notes on assessment (study guides; how instructors grade exams, etc.)

1 Overview for Instructors □

Video: Overview for Instructors [13:39]
Slides: Overview for Instructors

2 Overview for Students □

Course orientation: Some norms, expectations and tips
Video: Overview of the course [14:17]
Slides: Overview of the course

3 Fundamental issues in Software Testing □

Section Notes: Fundamental Issues in Software Testing
Quiz on the first lectures (up to oracles)
Quiz on complete testing
Submit the Oracles pretest

Latest News

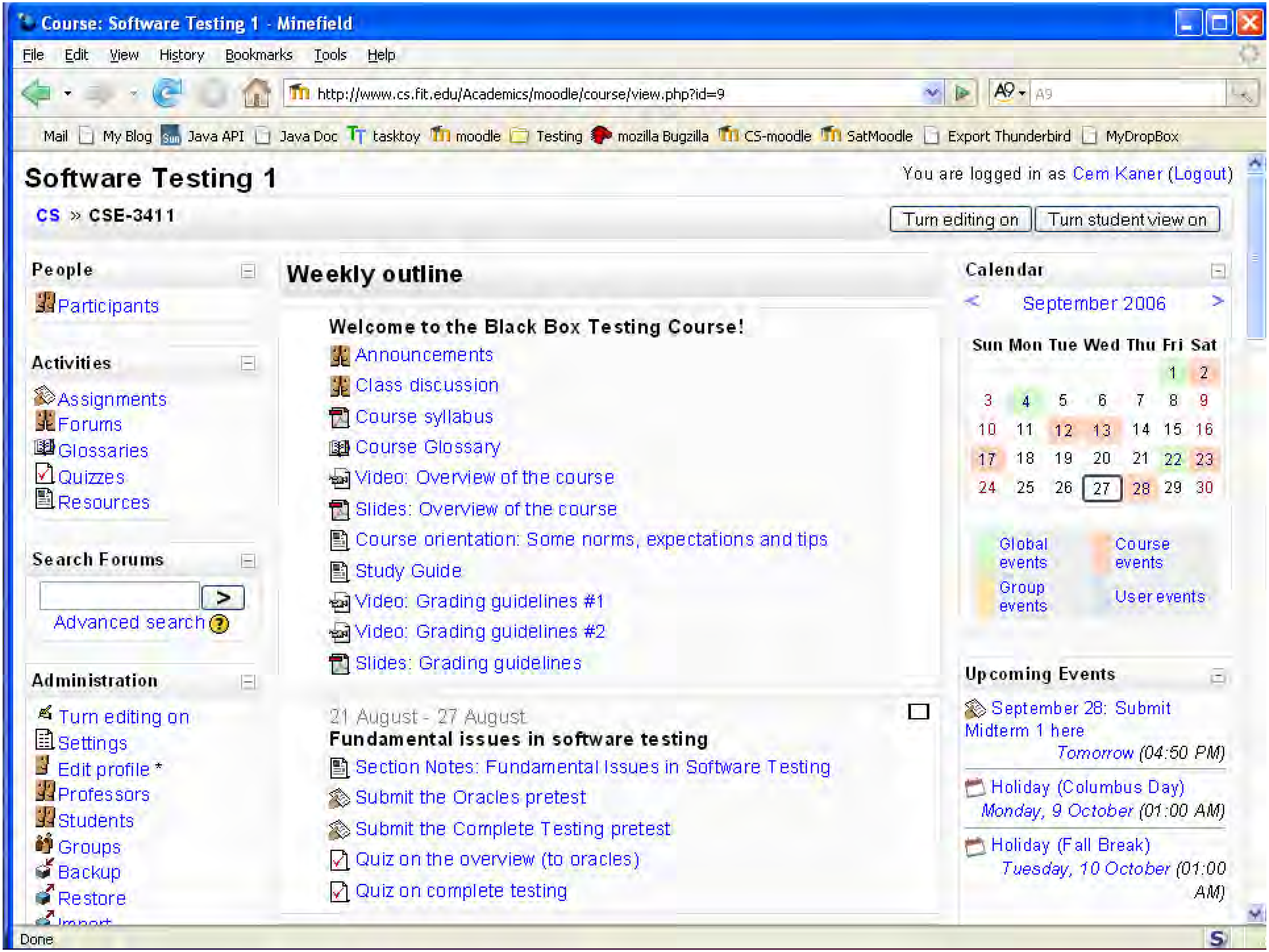
Add a new topic...
(No news has been posted yet)

Upcoming Events

There are no upcoming events

Go to calendar...
New Event...

Recent Activity


Course: Software Testing 1 - Minefield
File Edit View History Bookmarks Tools Help
http://www.cs.fit.edu/Academics/moodle/course/view.php?id=9
A9
Mail My Blog Java API Java Doc tasktoy moodle Testing mozilla Bugzilla CS-moodle SatMoodle Export Thunderbird MyDropBox
Turn editing on
Turn student view on
>
Done

Software Testing 1

You are logged in as Cem Kaner (Logout)

CS » CSE-3411

People

Participants

Activities

Assignments
Forums
Glossaries
Quizzes
Resources

Search Forums

Advanced search

Administration

Turn editing on
Settings
Edit profile *
Professors
Students
Groups
Backup
Restore
Import

Weekly outline

Welcome to the Black Box Testing Course!

Announcements
Class discussion
Course syllabus
Course Glossary
Video: Overview of the course
Slides: Overview of the course
Course orientation: Some norms, expectations and tips
Study Guide
Video: Grading guidelines #1
Video: Grading guidelines #2
Slides: Grading guidelines

21 August - 27 August

Fundamental issues in software testing

Section Notes: Fundamental Issues in Software Testing
Submit the Oracles pretest
Submit the Complete Testing pretest
Quiz on the overview (to oracles)
Quiz on complete testing

Calendar

< September 2006 >

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

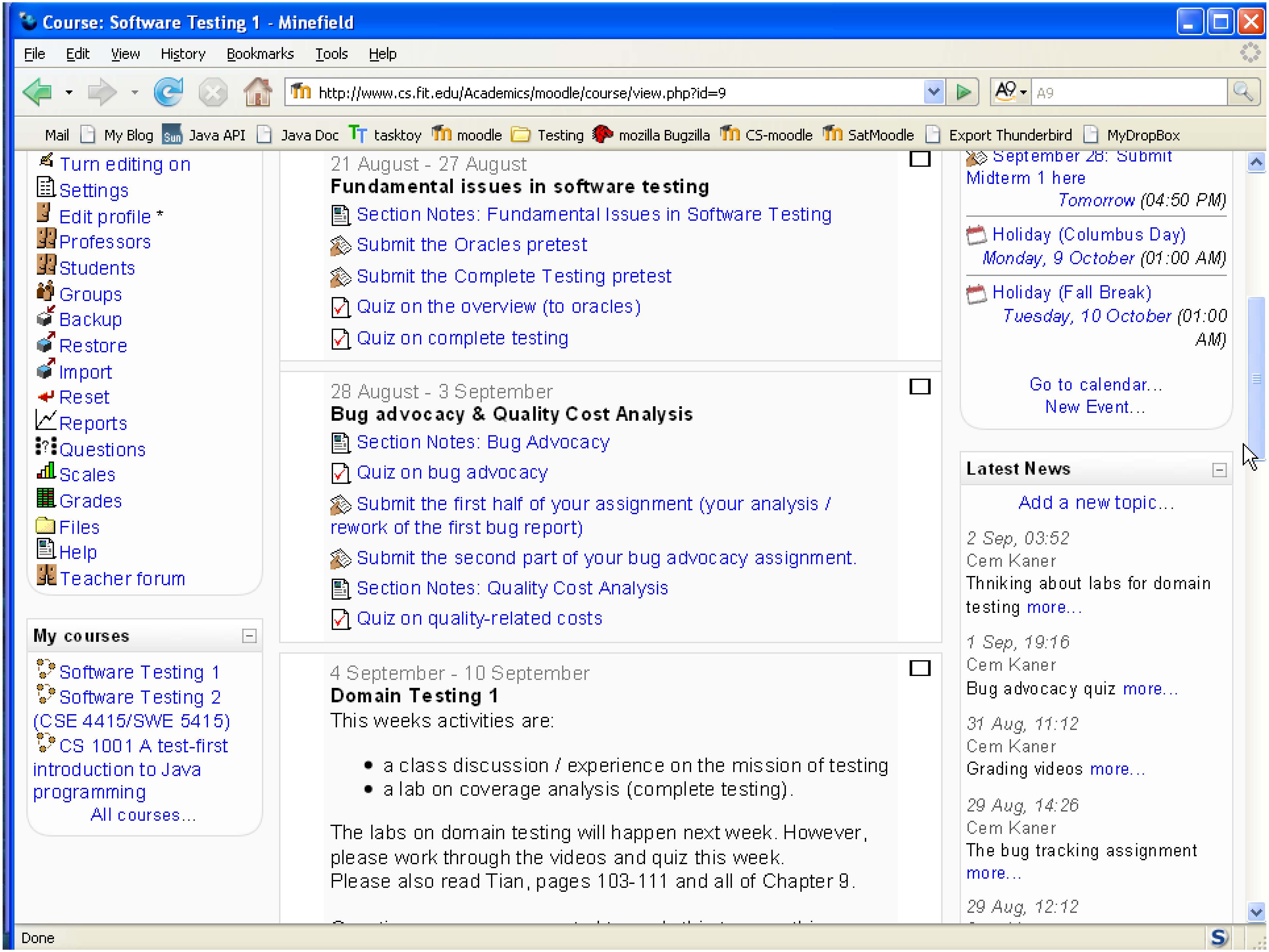
Global events
Course events
Group events
User events

Upcoming Events

September 28: Submit Midterm 1 here
Tomorrow (04:50 PM)

Holiday (Columbus Day)
Monday, 9 October (01:00 AM)

Holiday (Fall Break)
Tuesday, 10 October (01:00 AM)



- Turn editing on
- Settings
- Edit profile *
- Professors
- Students
- Groups
- Backup
- Restore
- Import
- Reset
- Reports
- Questions
- Scales
- Grades
- Files
- Help
- Teacher forum

My courses

- Software Testing 1
- Software Testing 2 (CSE 4415/SWE 5415)
- CS 1001 A test-first introduction to Java programming

All courses...

21 August - 27 August

Fundamental issues in software testing

- Section Notes: Fundamental Issues in Software Testing
- Submit the Oracles pretest
- Submit the Complete Testing pretest
- Quiz on the overview (to oracles)
- Quiz on complete testing

28 August - 3 September

Bug advocacy & Quality Cost Analysis

- Section Notes: Bug Advocacy
- Quiz on bug advocacy
- Submit the first half of your assignment (your analysis / rework of the first bug report)
- Submit the second part of your bug advocacy assignment.
- Section Notes: Quality Cost Analysis
- Quiz on quality-related costs

4 September - 10 September

Domain Testing 1

This weeks activities are:

- a class discussion / experience on the mission of testing
- a lab on coverage analysis (complete testing).

The labs on domain testing will happen next week. However, please work through the videos and quiz this week.
Please also read Tian, pages 103-111 and all of Chapter 9.

September 28: Submit Midterm 1 here
Tomorrow (04:50 PM)

Holiday (Columbus Day)
Monday, 9 October (01:00 AM)

Holiday (Fall Break)
Tuesday, 10 October (01:00 AM)

Go to calendar...
New Event...

Latest News

Add a new topic...

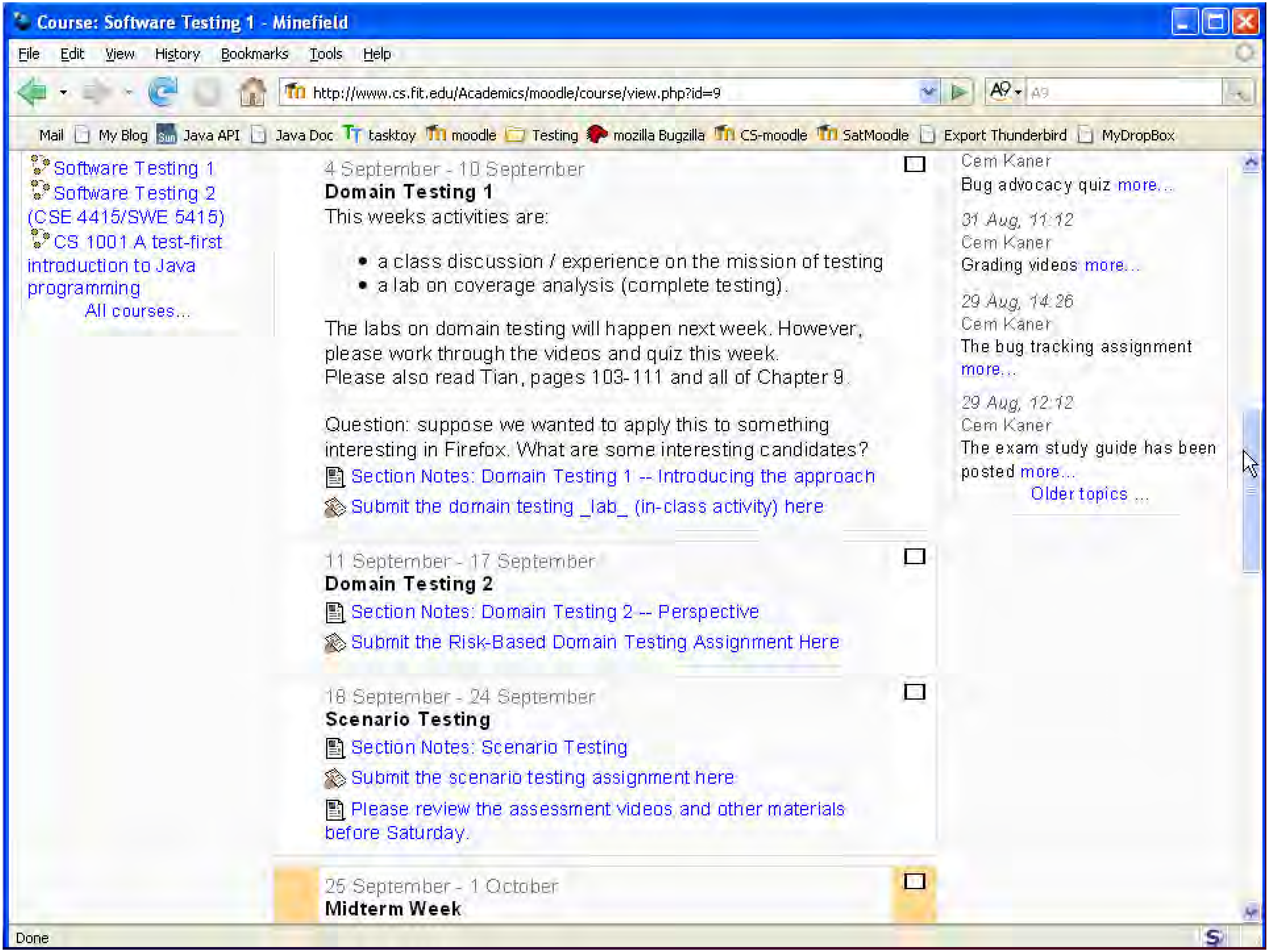
2 Sep, 03:52
Cem Kaner
Thniking about labs for domain testing more...

1 Sep, 19:16
Cem Kaner
Bug advocacy quiz more...

31 Aug, 11:12
Cem Kaner
Grading videos more...

29 Aug, 14:26
Cem Kaner
The bug tracking assignment more...

29 Aug, 12:12



Software Testing 1
Software Testing 2 (CSE 4415/SWE 5415)
CS 1001 A test-first introduction to Java programming
All courses...

4 September - 10 September ☐

Domain Testing 1

This weeks activities are:

- a class discussion / experience on the mission of testing
- a lab on coverage analysis (complete testing).

The labs on domain testing will happen next week. However, please work through the videos and quiz this week.
Please also read Tian, pages 103-111 and all of Chapter 9.

Question: suppose we wanted to apply this to something interesting in Firefox. What are some interesting candidates?

Section Notes: Domain Testing 1 -- Introducing the approach

Submit the domain testing _lab_ (in-class activity) here

11 September - 17 September ☐

Domain Testing 2

Section Notes: Domain Testing 2 -- Perspective

Submit the Risk-Based Domain Testing Assignment Here

18 September - 24 September ☐

Scenario Testing

Section Notes: Scenario Testing

Submit the scenario testing assignment here

Please review the assessment videos and other materials before Saturday.

25 September - 1 October ☐

Midterm Week

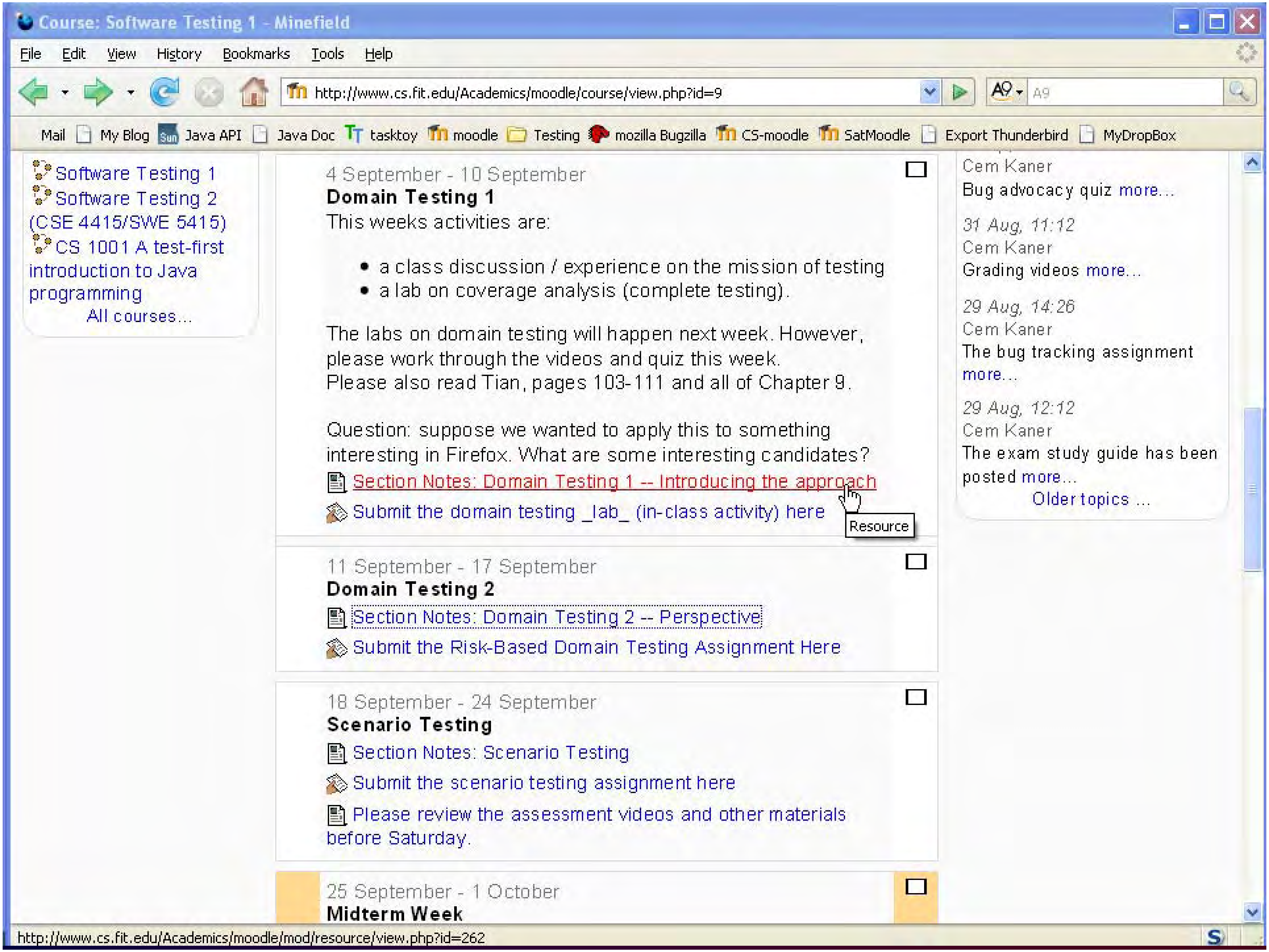
Cem Kaner
Bug advocacy quiz more...

31 Aug, 11:12
Cem Kaner
Grading videos more...

29 Aug, 14:26
Cem Kaner
The bug tracking assignment more...

29 Aug, 12:12
Cem Kaner
The exam study guide has been posted more...

Older topics ...


Course: Software Testing 1 - Minefield
File
Edit
View
History
Bookmarks
Tools
Help
http://www.cs.fit.edu/Academics/moodle/course/view.php?id=9
A9
Mail
My Blog
Java API
Java Doc
tasktoy
moodle
Testing
mozilla Bugzilla
CS-moodle
SatMoodle
Export Thunderbird
MyDropBox
Resource
http://www.cs.fit.edu/Academics/moodle/mod/resource/view.php?id=262

Software Testing 1
Software Testing 2 (CSE 4415/SWE 5415)
CS 1001 A test-first introduction to Java programming
All courses...

4 September - 10 September

Domain Testing 1

This weeks activities are:

- a class discussion / experience on the mission of testing
- a lab on coverage analysis (complete testing).

The labs on domain testing will happen next week. However, please work through the videos and quiz this week.
Please also read Tian, pages 103-111 and all of Chapter 9.

Question: suppose we wanted to apply this to something interesting in Firefox. What are some interesting candidates?

Section Notes: Domain Testing 1 -- Introducing the approach

Submit the domain testing _lab_ (in-class activity) here

11 September - 17 September

Domain Testing 2

Section Notes: Domain Testing 2 -- Perspective

Submit the Risk-Based Domain Testing Assignment Here

18 September - 24 September

Scenario Testing

Section Notes: Scenario Testing

Submit the scenario testing assignment here

Please review the assessment videos and other materials before Saturday.

25 September - 1 October

Midterm Week

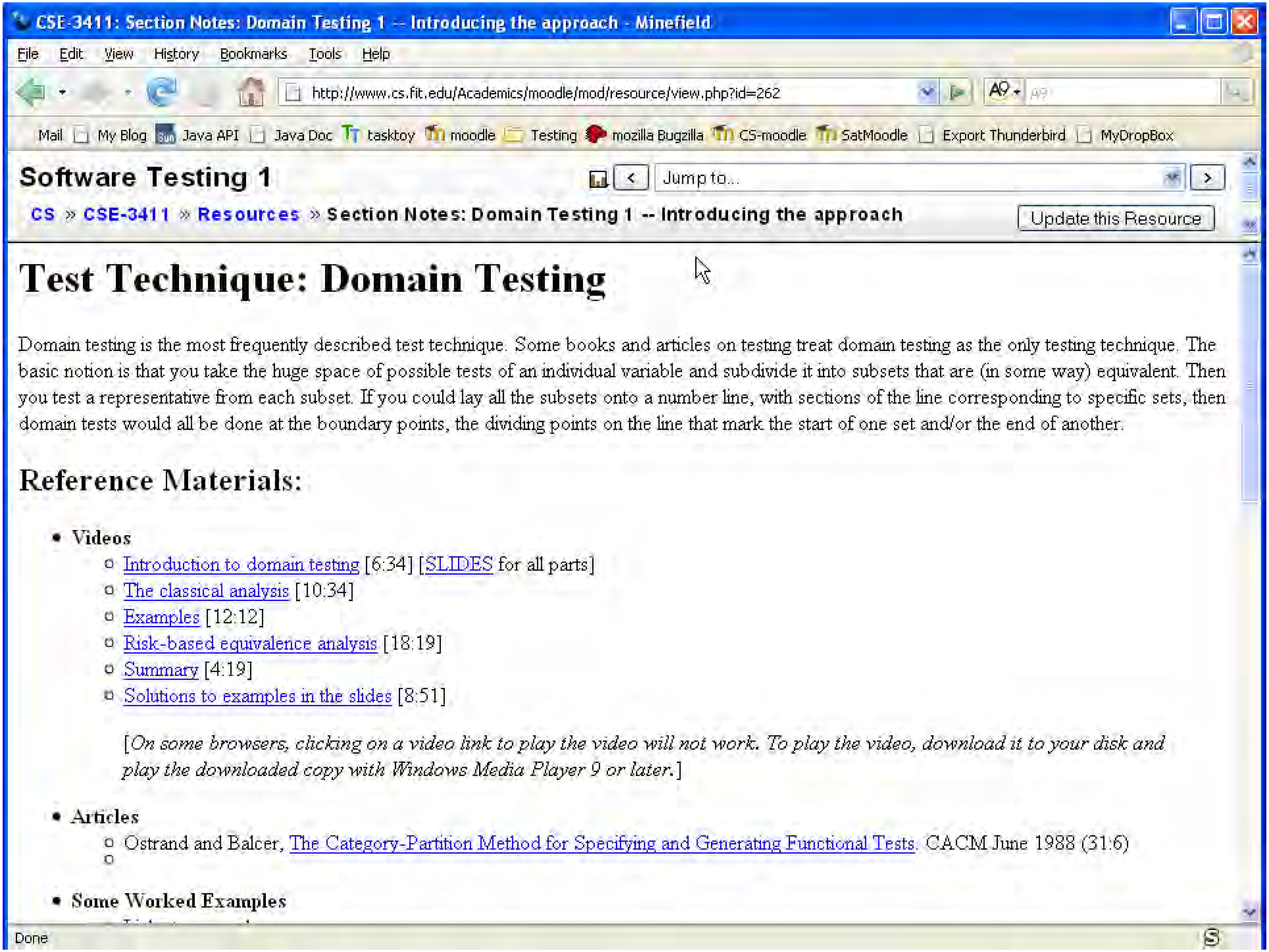
Cem Kaner
Bug advocacy quiz more...

31 Aug, 11:12
Cem Kaner
Grading videos more...

29 Aug, 14:26
Cem Kaner
The bug tracking assignment more...

29 Aug, 12:12
Cem Kaner
The exam study guide has been posted more...

Older topics ...


CSE-3411: Section Notes: Domain Testing 1 -- Introducing the approach - Minefield
File Edit View History Bookmarks Tools Help
http://www.cs.fit.edu/Academics/moodle/mod/resource/view.php?id=262
Mail My Blog Java API Java Doc tasktoy moodle Testing mozilla Bugzilla CS-moodle SatMoodle Export Thunderbird MyDropBox
Jump to...
Update this Resource
Done

Software Testing 1

CS » CSE-3411 » Resources » Section Notes: Domain Testing 1 -- Introducing the approach

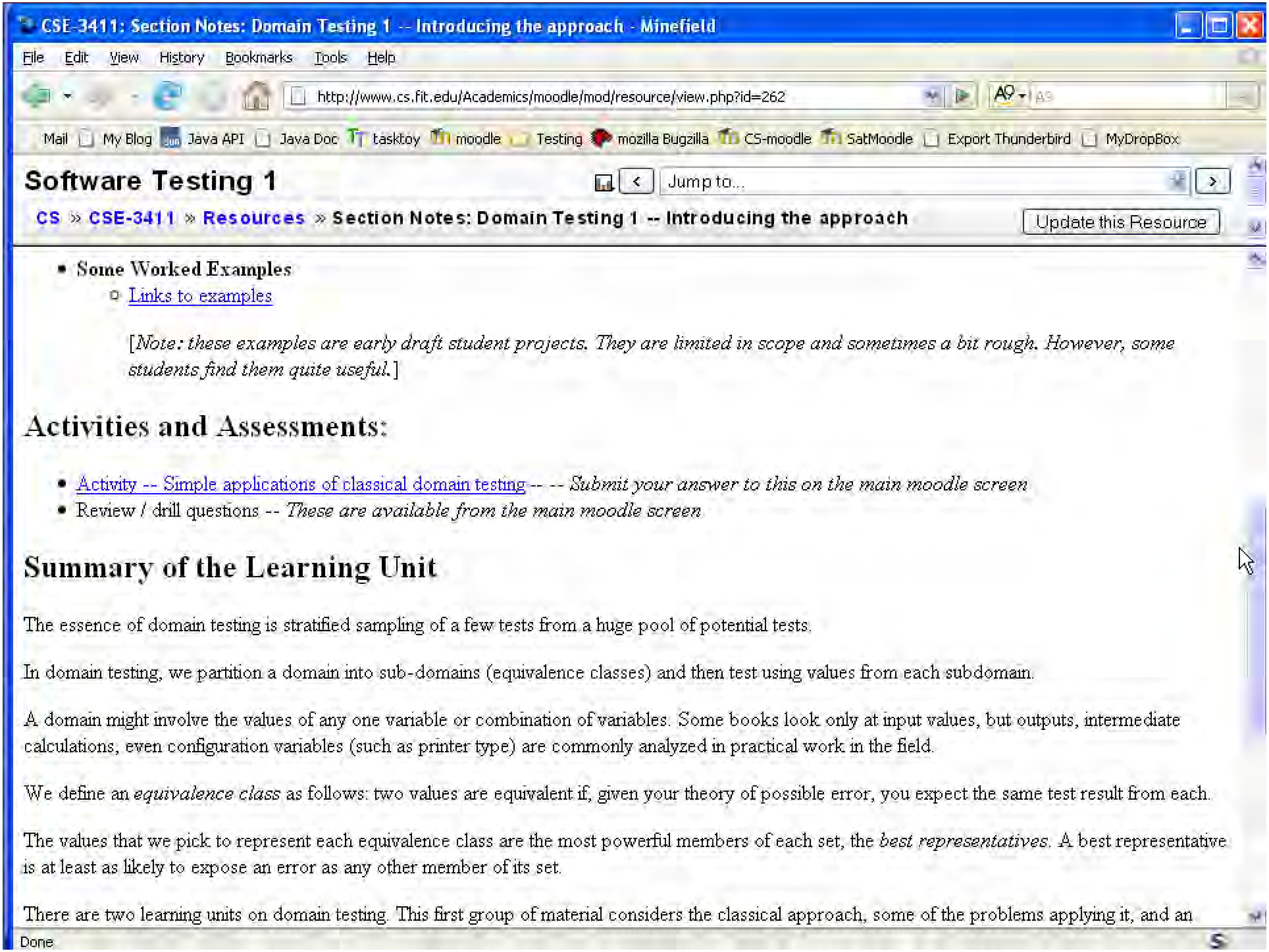
Test Technique: Domain Testing

Domain testing is the most frequently described test technique. Some books and articles on testing treat domain testing as the only testing technique. The basic notion is that you take the huge space of possible tests of an individual variable and subdivide it into subsets that are (in some way) equivalent. Then you test a representative from each subset. If you could lay all the subsets onto a number line, with sections of the line corresponding to specific sets, then domain tests would all be done at the boundary points, the dividing points on the line that mark the start of one set and/or the end of another.

Reference Materials:

- **Videos**
  - Introduction to domain testing [6:34] [SLIDES for all parts]
  - The classical analysis [10:34]
  - Examples [12:12]
  - Risk-based equivalence analysis [18:19]
  - Summary [4:19]
  - Solutions to examples in the slides [8:51]

  [On some browsers, clicking on a video link to play the video will not work. To play the video, download it to your disk and play the downloaded copy with Windows Media Player 9 or later.]
- **Articles**
  - Ostrand and Balcer, The Category-Partition Method for Specifying and Generating Functional Tests. CACM June 1988 (31:6)
  -
- **Some Worked Examples**



CSE-3411: Section Notes: Domain Testing 1 -- Introducing the approach - Minefield
File Edit View History Bookmarks Tools Help
http://www.cs.fit.edu/Academics/moodle/mod/resource/view.php?id=262
Mail My Blog Java API Java Doc tasktoy moodle Testing mozilla Bugzilla CS-moodle SatMoodle Export Thunderbird MyDropBox
Jump to...
Update this Resource
Done

Software Testing 1

CS » CSE-3411 » Resources » Section Notes: Domain Testing 1 -- Introducing the approach

- Some Worked Examples
  - Links to examples

    [*Note: these examples are early draft student projects. They are limited in scope and sometimes a bit rough. However, some students find them quite useful.*]

Activities and Assessments:

- Activity -- Simple applications of classical domain testing -- -- *Submit your answer to this on the main moodle screen*
- Review / drill questions -- *These are available from the main moodle screen*

Summary of the Learning Unit

The essence of domain testing is stratified sampling of a few tests from a huge pool of potential tests.

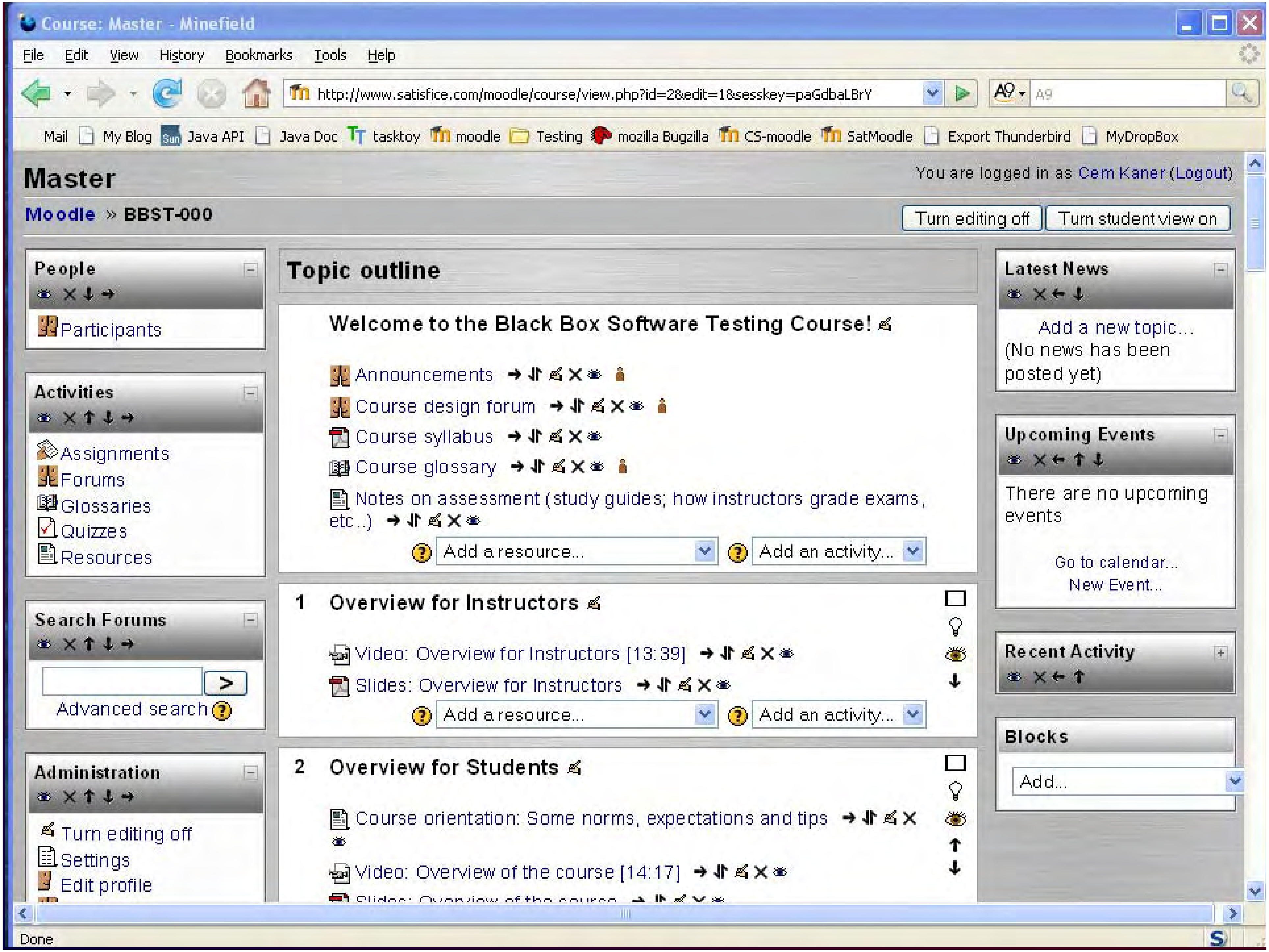
In domain testing, we partition a domain into sub-domains (equivalence classes) and then test using values from each subdomain.

A domain might involve the values of any one variable or combination of variables. Some books look only at input values, but outputs, intermediate calculations, even configuration variables (such as printer type) are commonly analyzed in practical work in the field.

We define an *equivalence class* as follows: two values are equivalent if, given your theory of possible error, you expect the same test result from each.

The values that we pick to represent each equivalence class are the most powerful members of each set, the *best representatives*. A best representative is at least as likely to expose an error as any other member of its set.

There are two learning units on domain testing. This first group of material considers the classical approach, some of the problems applying it, and an


Course: Master - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/course/view.php?id=2&edit=1&sesskey=paGdbaLBrY
A9
Mail My Blog Java API Java Doc tasktoy moodle Testing mozilla Bugzilla CS-moodle SatMoodle Export Thunderbird MyDropBox
Turn editing off
Turn student view on
Add a resource...
Add an activity...
Add...
Done

Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000

People

Participants

Activities

Assignments
Forums
Glossaries
Quizzes
Resources

Search Forums

Advanced search

Administration

Turn editing off
Settings
Edit profile

Topic outline

Welcome to the Black Box Software Testing Course!

Announcements
Course design forum
Course syllabus
Course glossary
Notes on assessment (study guides; how instructors grade exams, etc..)

1 Overview for Instructors

Video: Overview for Instructors [13:39]
Slides: Overview for Instructors

2 Overview for Students

Course orientation: Some norms, expectations and tips
Video: Overview of the course [14:17]
Slides: Overview of the course

Latest News

Add a new topic...
(No news has been posted yet)

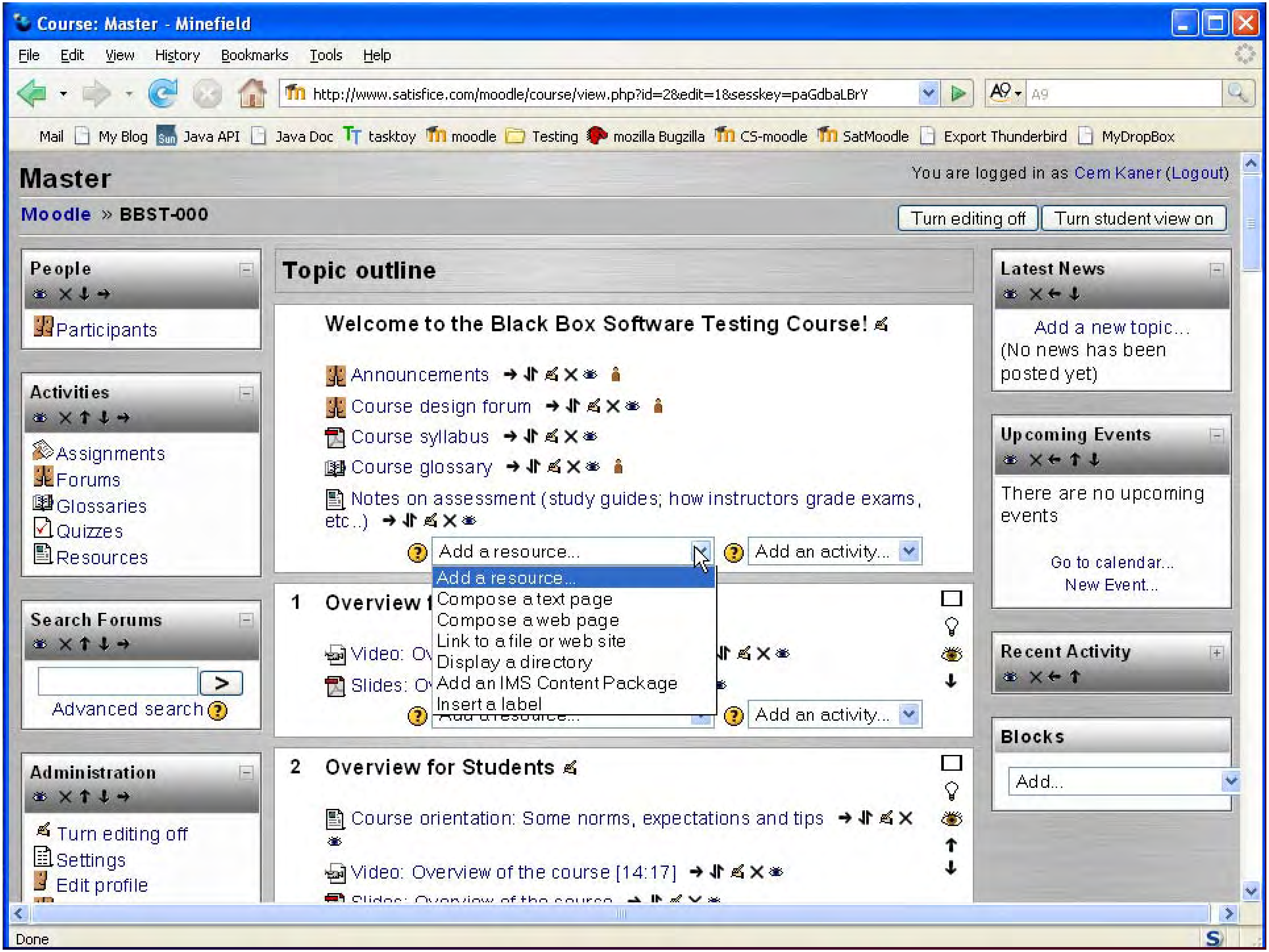
Upcoming Events

There are no upcoming events

Go to calendar...
New Event...

Recent Activity

Blocks


Course: Master - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/course/view.php?id=2&edit=1&sesskey=paGdbaLBrY
Mail My Blog Java API Java Doc tasktoy moodle Testing mozilla Bugzilla CS-moodle SatMoodle Export Thunderbird MyDropBox
Turn editing off
Turn student view on
Add a resource...
Compose a text page
Compose a web page
Link to a file or web site
Display a directory
Add an IMS Content Package
Insert a label
Add an activity...
Add...
Done

Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000

People

Participants

Activities

Assignments
Forums
Glossaries
Quizzes
Resources

Search Forums

Advanced search

Administration

Turn editing off
Settings
Edit profile

Topic outline

Welcome to the Black Box Software Testing Course!

Announcements
Course design forum
Course syllabus
Course glossary
Notes on assessment (study guides; how instructors grade exams, etc..)

1 Overview f

Video: Ov
Slides: O

2 Overview for Students

Course orientation: Some norms, expectations and tips
Video: Overview of the course [14:17]
Slides: Overview of the course

Latest News

Add a new topic...
(No news has been posted yet)

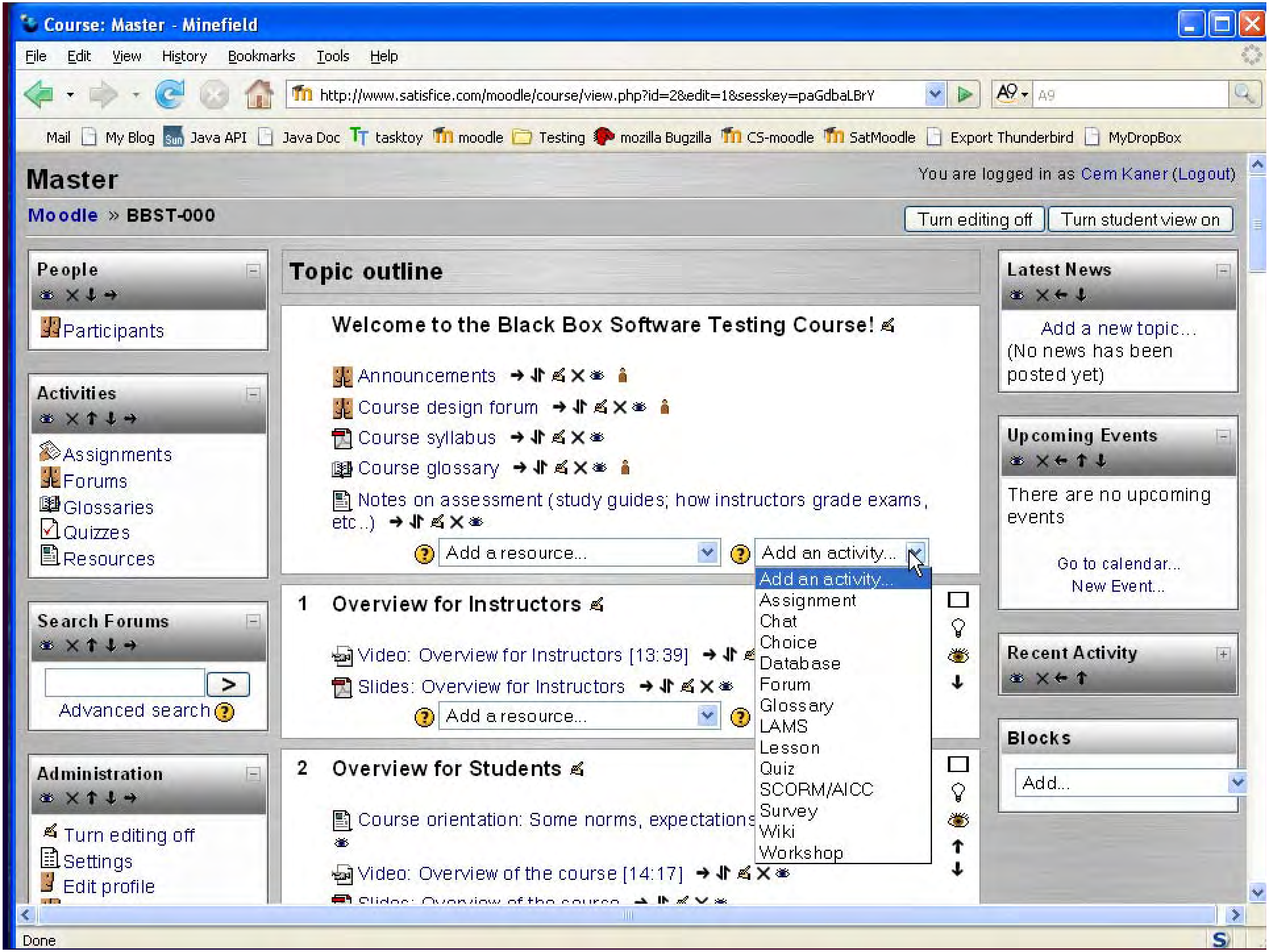
Upcoming Events

There are no upcoming events

Go to calendar...
New Event...

Recent Activity

Blocks



Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000

People

Participants

Activities

Assignments

Forums

Glossaries

Quizzes

Resources

Search Forums

Advanced search

Administration

Turn editing off

Settings

Edit profile

Topic outline

Welcome to the Black Box Software Testing Course!

Announcements

Course design forum

Course syllabus

Course glossary

Notes on assessment (study guides; how instructors grade exams, etc..)

1 Overview for Instructors

Video: Overview for Instructors [13:39]

Slides: Overview for Instructors

2 Overview for Students

Course orientation: Some norms, expectations

Video: Overview of the course [14:17]

Slides: Overview of the course

Latest News

Add a new topic...

(No news has been posted yet)

Upcoming Events

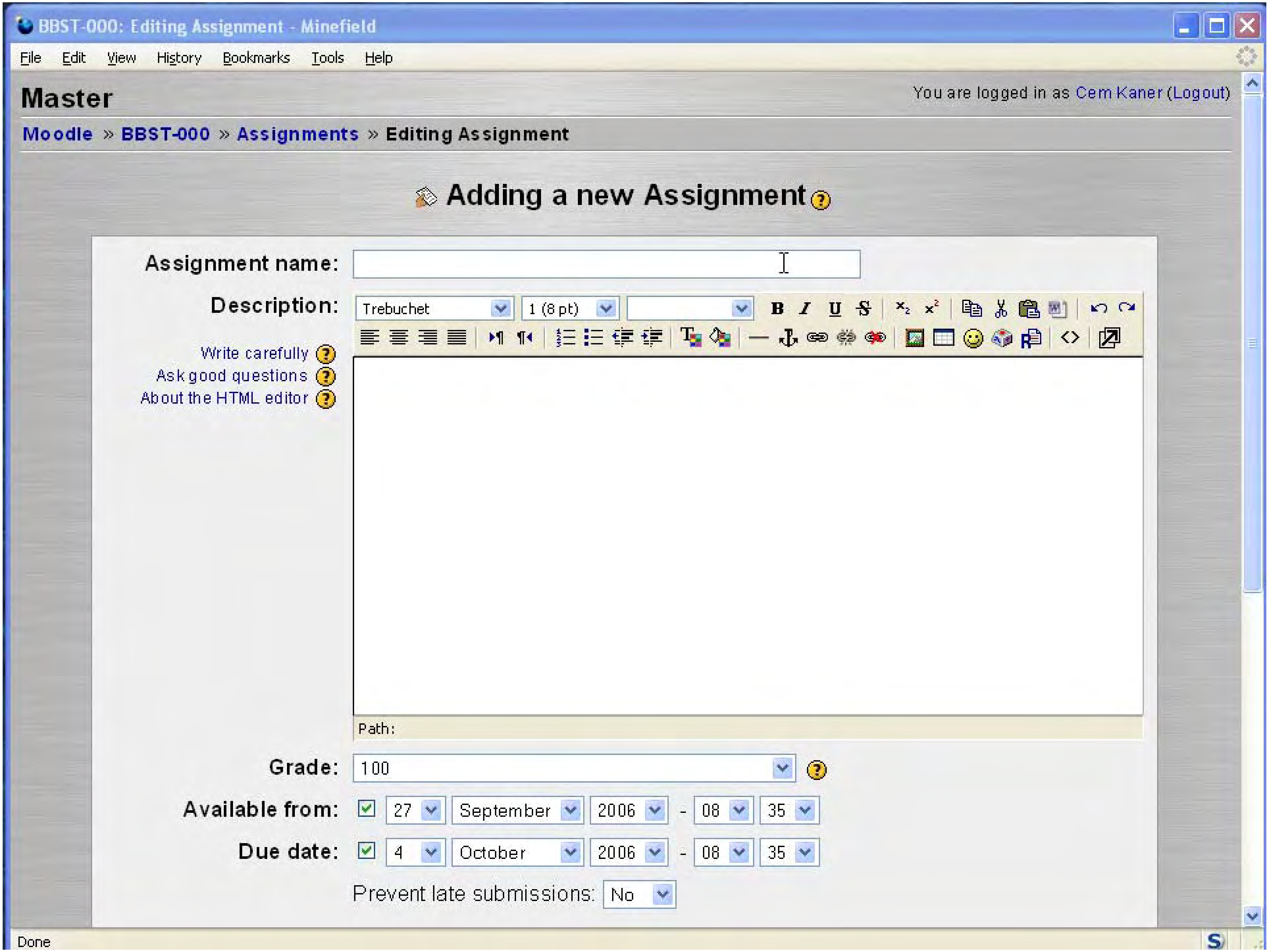
There are no upcoming events

Go to calendar...

New Event...

Recent Activity

Blocks



BBST-000: Editing Assignment - Minefield
File Edit View History Bookmarks Tools Help

Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000 » Assignments » Editing Assignment

Adding a new Assignment

Assignment name:

Description:

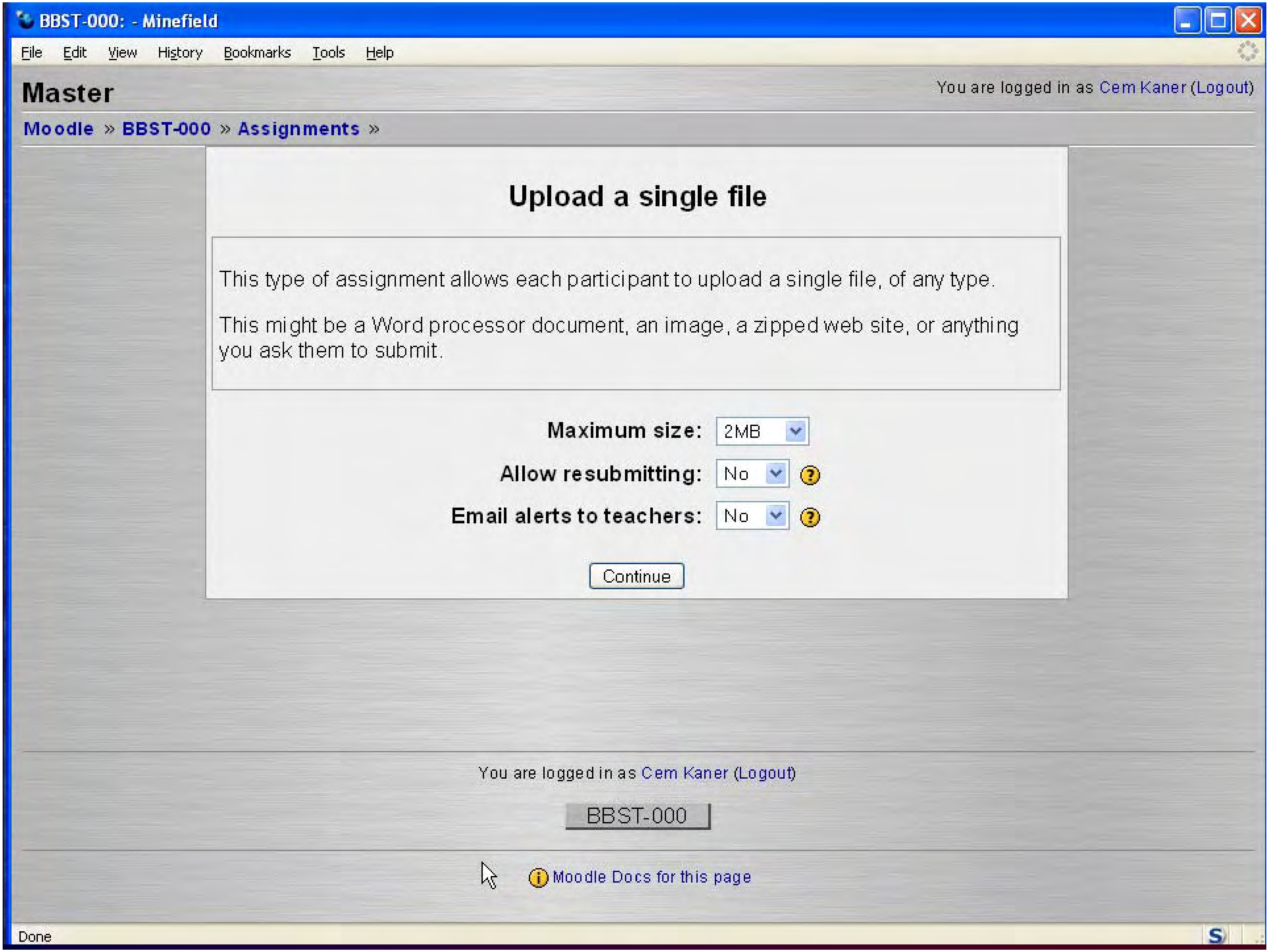
Trebuchet
1 (8 pt)
Write carefully
Ask good questions
About the HTML editor
Path:

Grade: 100

Available from: ☑ 27 September 2006 - 08 35

Due date: ☑ 4 October 2006 - 08 35

Prevent late submissions: No



BBST-000: - Minefield
File Edit View History Bookmarks Tools Help
Master
You are logged in as Cem Kaner (Logout)
Moodle » BBST-000 » Assignments »

Upload a single file

This type of assignment allows each participant to upload a single file, of any type.

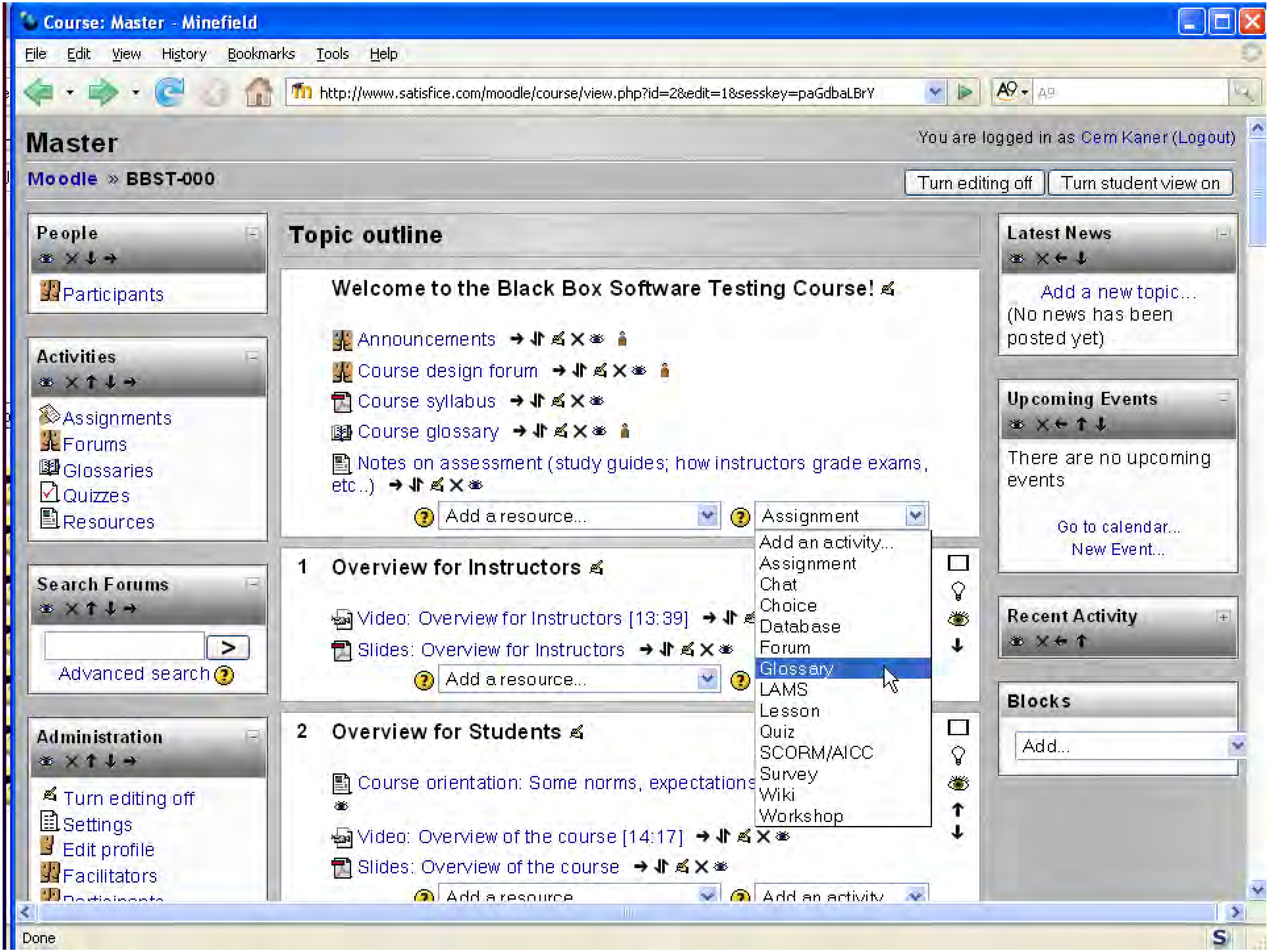
This might be a Word processor document, an image, a zipped web site, or anything you ask them to submit.

Maximum size: 2MB

Allow resubmitting: No

Email alerts to teachers: No

Continue
You are logged in as Cem Kaner (Logout)
BBST-000
Moodle Docs for this page
Done


Course: Master - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/course/view.php?id=2&edit=1&sesskey=paGdbaLBrY
Turn editing off
Turn student view on
Done

Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000

People

Participants

Activities

Assignments
Forums
Glossaries
Quizzes
Resources

Search Forums

Advanced search

Administration

Turn editing off
Settings
Edit profile
Facilitators
Participants

Topic outline

Welcome to the Black Box Software Testing Course!

Announcements
Course design forum
Course syllabus
Course glossary
Notes on assessment (study guides; how instructors grade exams, etc..)

Add a resource...
Assignment

Add an activity...
Assignment
Chat
Choice
Database
Forum
Glossary
LAMS
Lesson
Quiz
SCORM/AICC
Survey
Wiki
Workshop

1 **Overview for Instructors**

Video: Overview for Instructors [13:39]
Slides: Overview for Instructors

Add a resource...

2 **Overview for Students**

Course orientation: Some norms, expectations
Video: Overview of the course [14:17]
Slides: Overview of the course

Add a resource...
Add an activity...

Latest News

Add a new topic...
(No news has been posted yet)

Upcoming Events

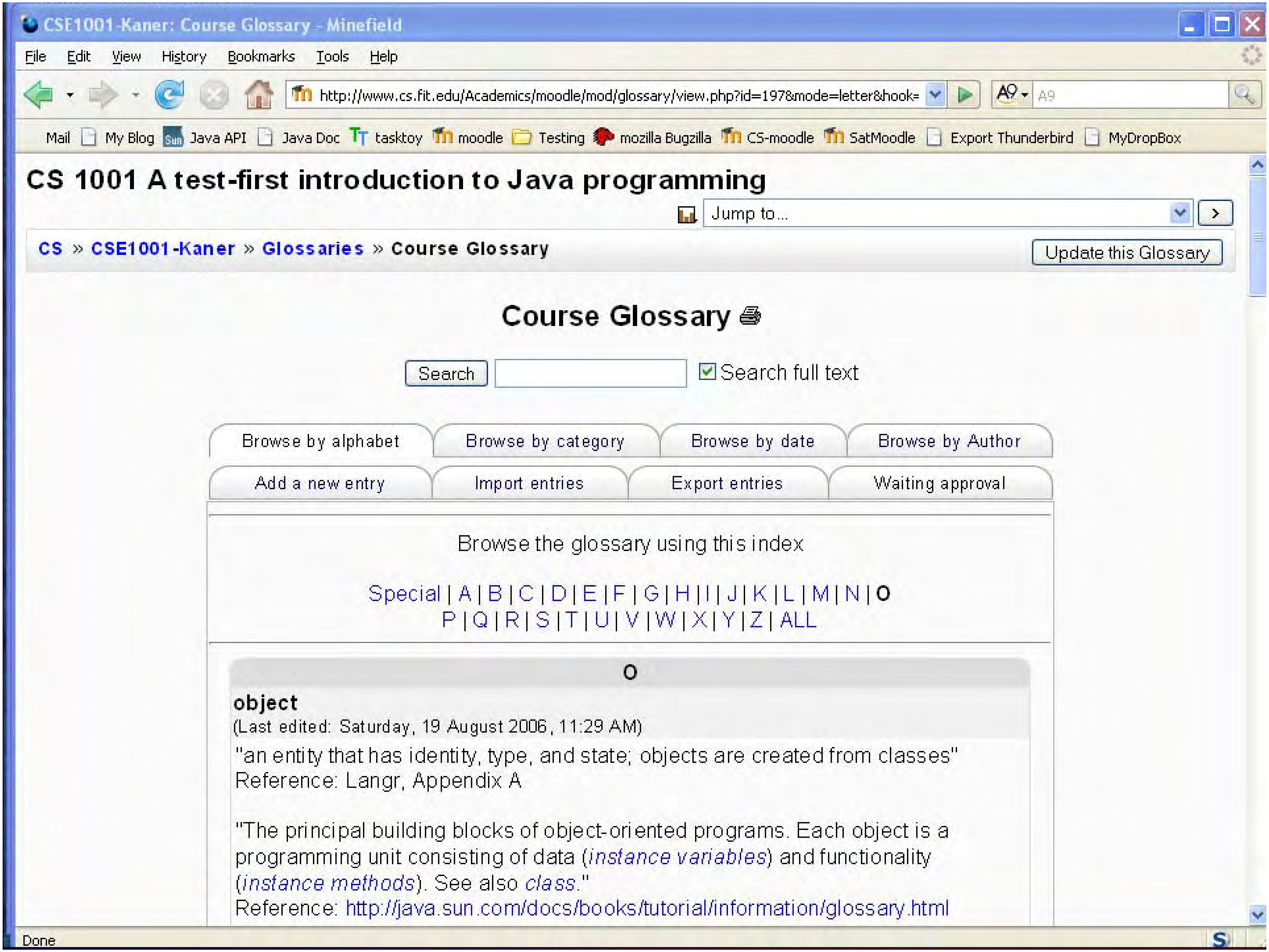
There are no upcoming events

Go to calendar...
New Event...

Recent Activity

Blocks

Add...


CSE1001-Kaner: Course Glossary - Minefield
File Edit View History Bookmarks Tools Help
http://www.cs.fit.edu/Academics/moodle/mod/glossary/view.php?id=197&mode=letter&hook=
Mail My Blog Java API Java Doc tasktoy moodle Testing mozilla Bugzilla CS-moodle SatMoodle Export Thunderbird MyDropBox
Jump to...
Update this Glossary
Search
☑ Search full text
Browse by alphabet
Browse by category
Browse by date
Browse by Author
Add a new entry
Import entries
Export entries
Waiting approval
Done

CS 1001 A test-first introduction to Java programming

CS » CSE1001-Kaner » Glossaries » Course Glossary

Course Glossary

Browse the glossary using this index

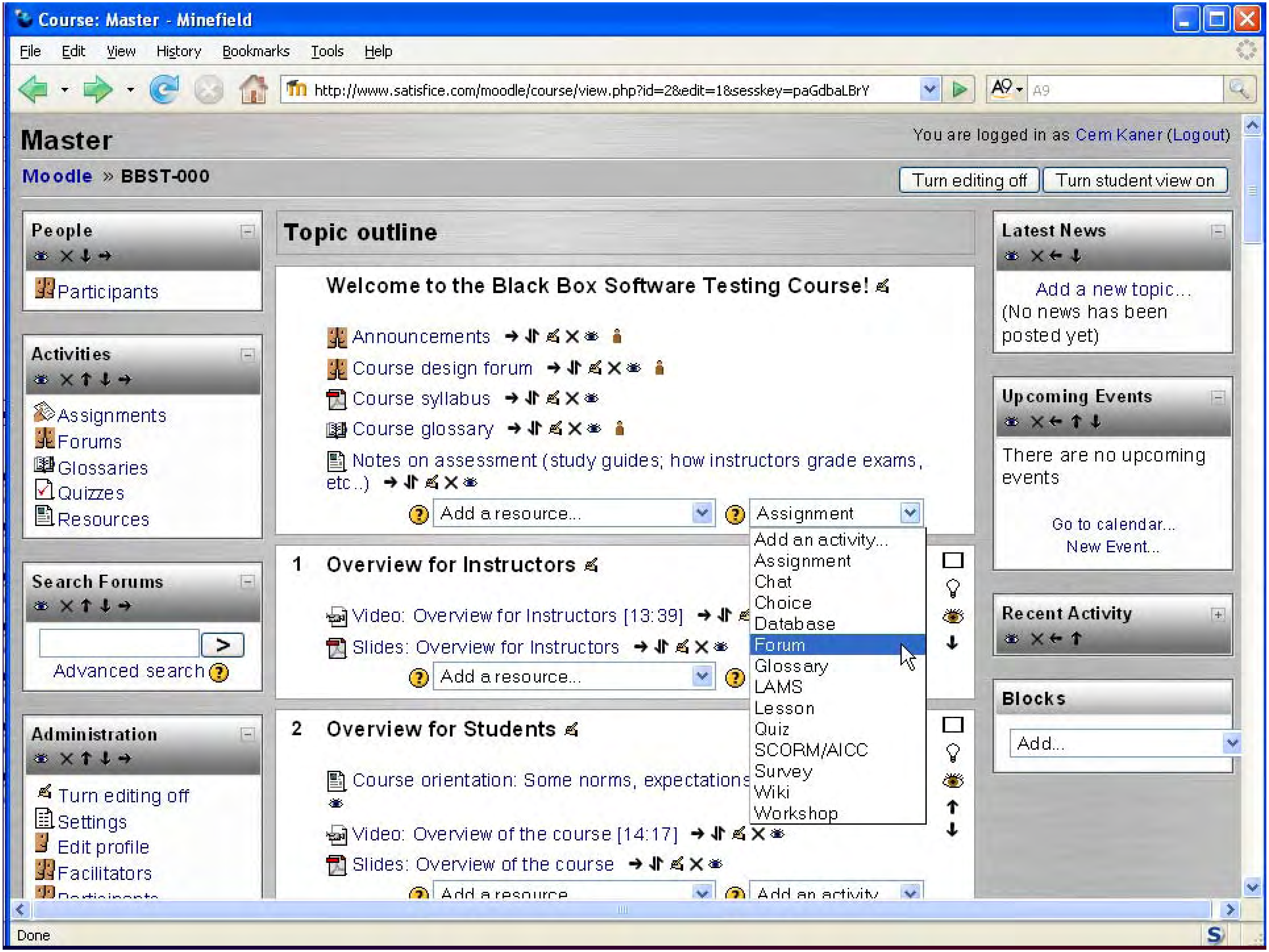
Special | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O
P | Q | R | S | T | U | V | W | X | Y | Z | ALL

O

object
(Last edited: Saturday, 19 August 2006, 11:29 AM)

"an entity that has identity, type, and state; objects are created from classes"
Reference: Langr, Appendix A

"The principal building blocks of object-oriented programs. Each object is a programming unit consisting of data (*instance variables*) and functionality (*instance methods*). See also *class*."
Reference: http://java.sun.com/docs/books/tutorial/information/glossary.html



Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000

People

Participants

Activities

Assignments
Forums
Glossaries
Quizzes
Resources

Search Forums

Advanced search

Administration

Turn editing off
Settings
Edit profile
Facilitators
Participants

Topic outline

Welcome to the Black Box Software Testing Course!

Announcements
Course design forum
Course syllabus
Course glossary
Notes on assessment (study guides; how instructors grade exams, etc..)

1 Overview for Instructors

Video: Overview for Instructors [13:39]
Slides: Overview for Instructors

2 Overview for Students

Course orientation: Some norms, expectations
Video: Overview of the course [14:17]
Slides: Overview of the course

Latest News

Add a new topic...
(No news has been posted yet)

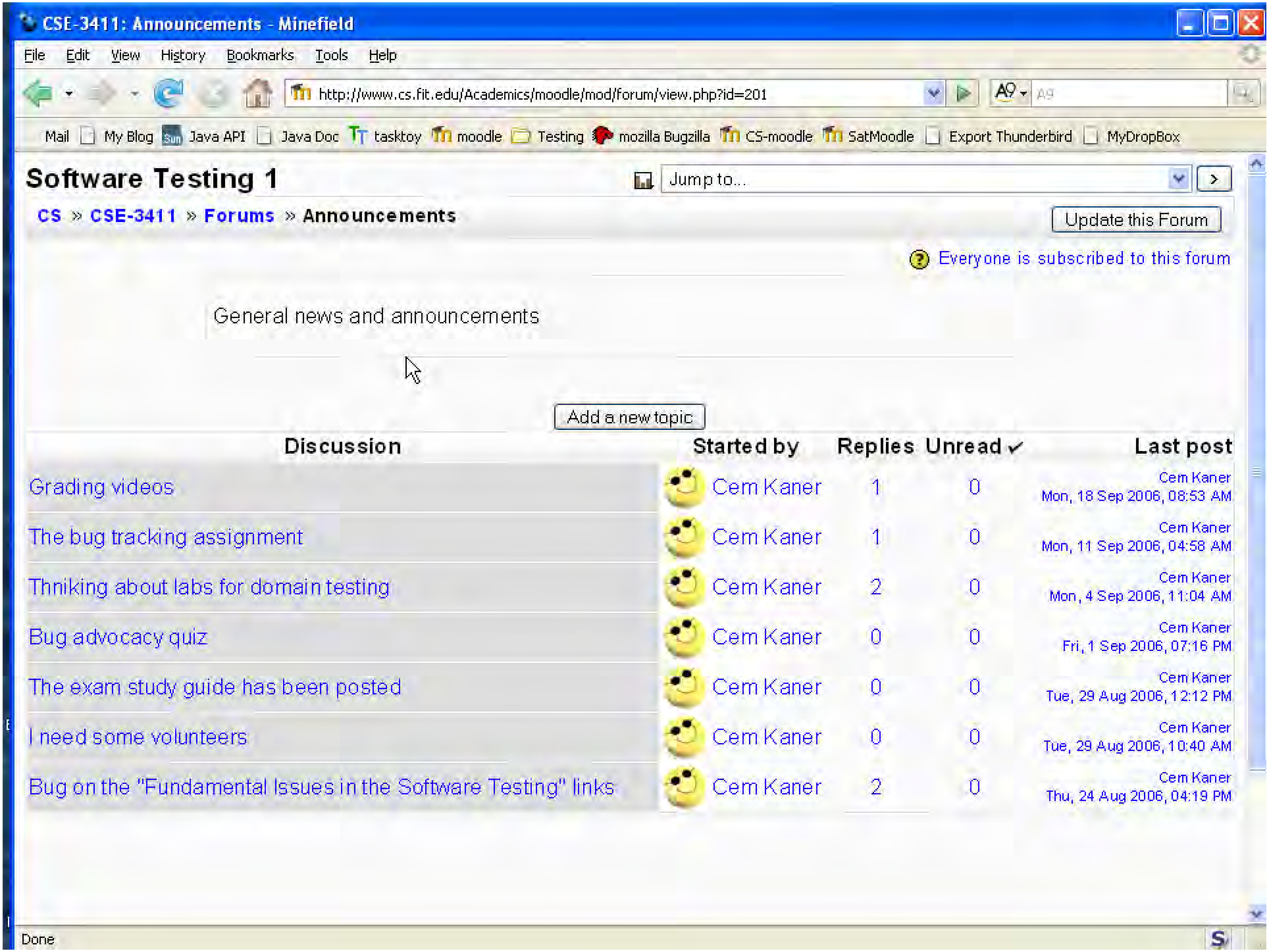
Upcoming Events

There are no upcoming events

Go to calendar...
New Event...

Recent Activity

Blocks



http://www.cs.fit.edu/Academics/moodle/mod/forum/view.php?id=201

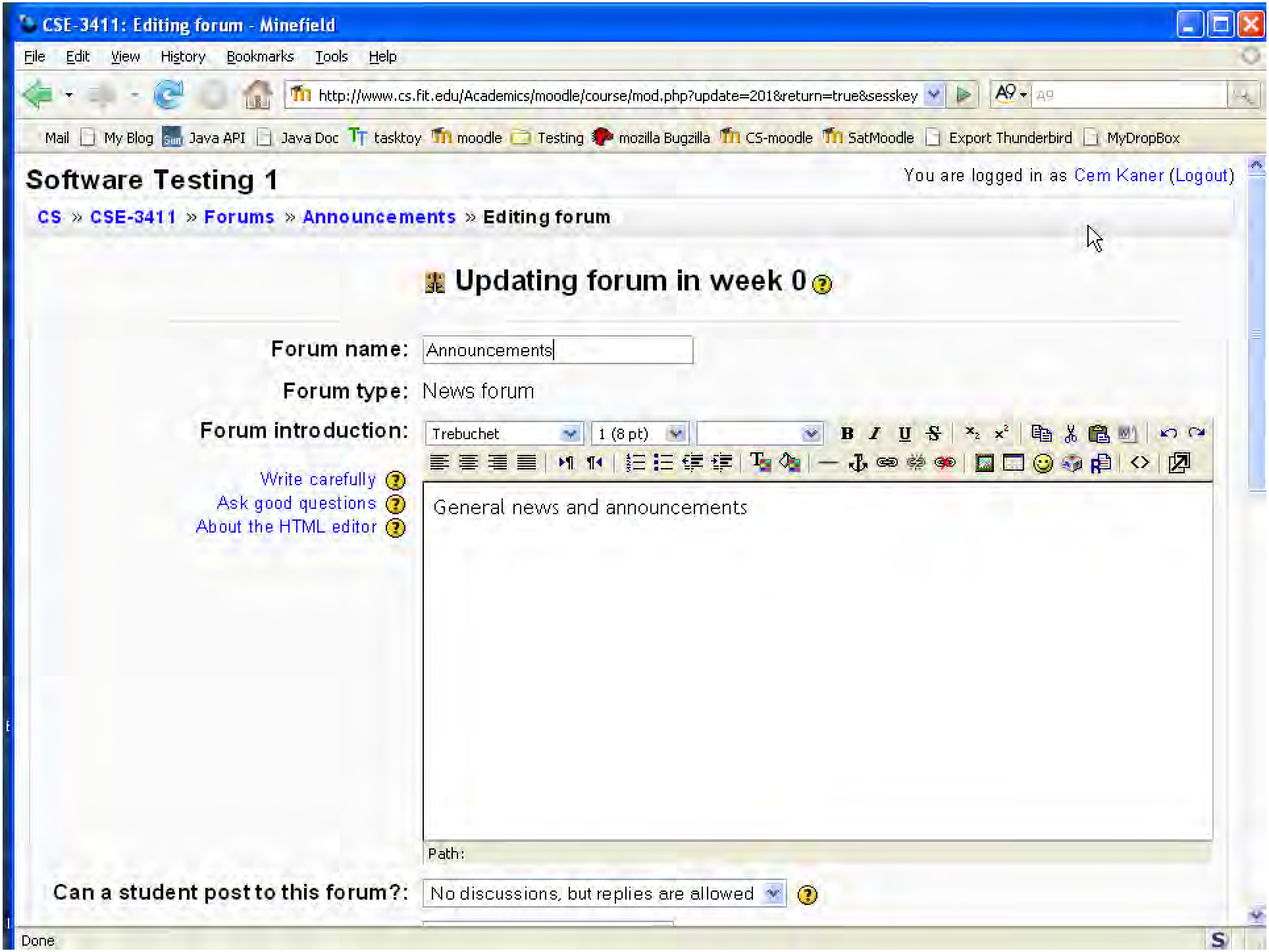
Software Testing 1

CS » CSE-3411 » Forums » Announcements

Everyone is subscribed to this forum

General news and announcements

Discussion	Started by	Replies	Unread ✓	Last post
Grading videos	Cem Kaner	1	0	Cem Kaner Mon, 18 Sep 2006, 08:53 AM
The bug tracking assignment	Cem Kaner	1	0	Cem Kaner Mon, 11 Sep 2006, 04:58 AM
Thniking about labs for domain testing	Cem Kaner	2	0	Cem Kaner Mon, 4 Sep 2006, 11:04 AM
Bug advocacy quiz	Cem Kaner	0	0	Cem Kaner Fri, 1 Sep 2006, 07:16 PM
The exam study guide has been posted	Cem Kaner	0	0	Cem Kaner Tue, 29 Aug 2006, 12:12 PM
I need some volunteers	Cem Kaner	0	0	Cem Kaner Tue, 29 Aug 2006, 10:40 AM
Bug on the "Fundamental Issues in the Software Testing" links	Cem Kaner	2	0	Cem Kaner Thu, 24 Aug 2006, 04:19 PM



Software Testing 1

You are logged in as Cem Kaner (Logout)

CS » CSE-3411 » Forums » Announcements » Editing forum

Updating forum in week 0

Forum name: Announcements

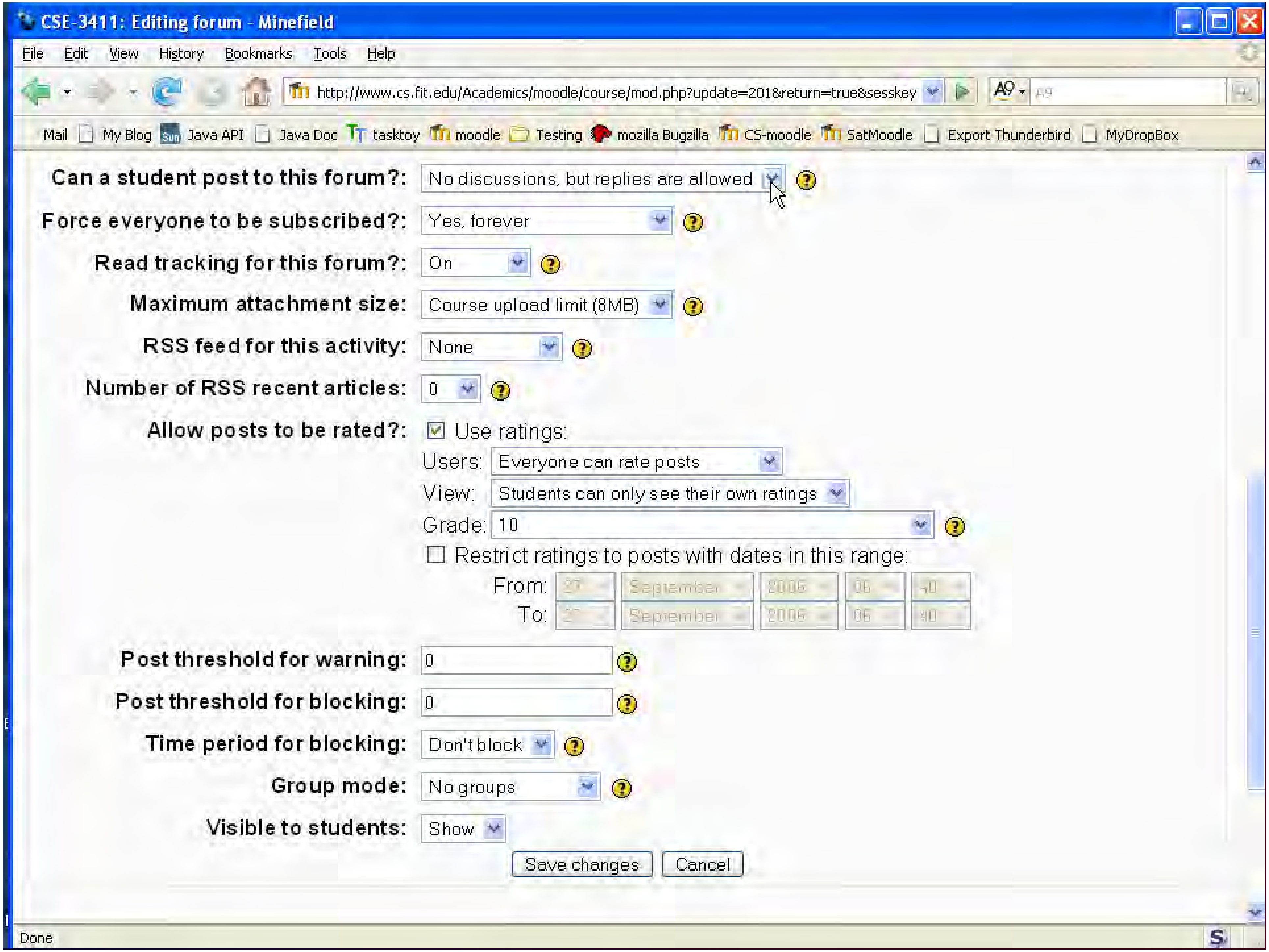
Forum type: News forum

Forum introduction: Trebuchet | 1 (8 pt)

General news and announcements

Path:

Can a student post to this forum?: No discussions, but replies are allowed


CSE-3411: Editing forum - Minefield
File Edit View History Bookmarks Tools Help
http://www.cs.fit.edu/Academics/moodle/course/mod.php?update=201&return=true&sesskey
Mail
My Blog
Java API
Java Doc
tasktoy
moodle
Testing
mozilla Bugzilla
CS-moodle
SatMoodle
Export Thunderbird
MyDropBox
Save changes
Cancel
Done

Can a student post to this forum?: No discussions, but replies are allowed

Force everyone to be subscribed?: Yes, forever

Read tracking for this forum?: On

Maximum attachment size: Course upload limit (8MB)

RSS feed for this activity: None

Number of RSS recent articles: 0

Allow posts to be rated?: ☑ Use ratings:

Users: Everyone can rate posts

View: Students can only see their own ratings

Grade: 10

□ Restrict ratings to posts with dates in this range:

From: 27 September 2006 06 40

To: 27 September 2006 06 40

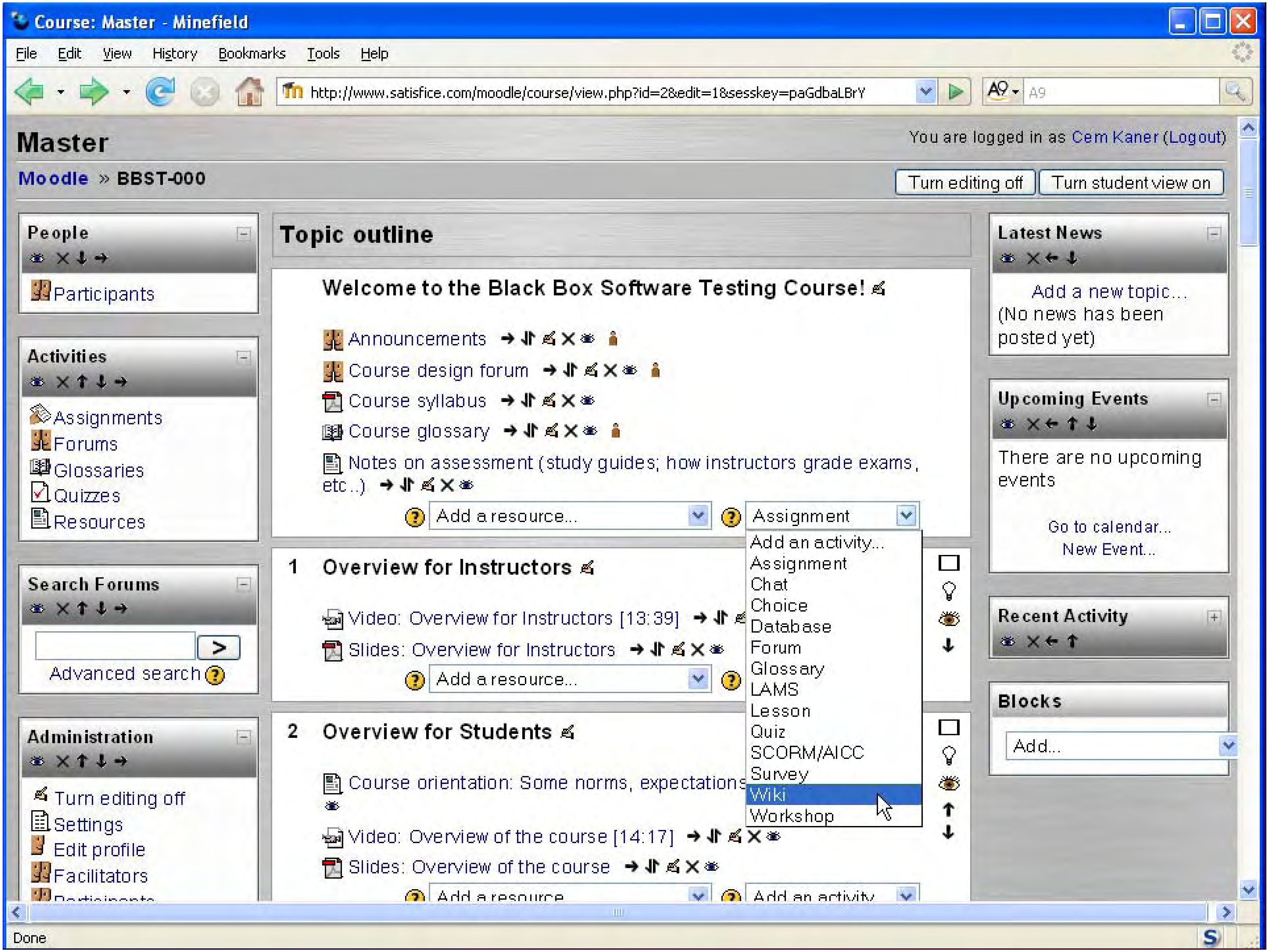
Post threshold for warning: 0

Post threshold for blocking: 0

Time period for blocking: Don't block

Group mode: No groups

Visible to students: Show



Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000

People

Participants

Activities

Assignments
Forums
Glossaries
Quizzes
Resources

Search Forums

Advanced search

Administration

Turn editing off
Settings
Edit profile
Facilitators
Participants

Topic outline

Welcome to the Black Box Software Testing Course!

Announcements
Course design forum
Course syllabus
Course glossary
Notes on assessment (study guides; how instructors grade exams, etc..)

1 Overview for Instructors

Video: Overview for Instructors [13:39]
Slides: Overview for Instructors

2 Overview for Students

Course orientation: Some norms, expectations
Video: Overview of the course [14:17]
Slides: Overview of the course

Add a resource... Add an activity

Latest News

Add a new topic...
(No news has been posted yet)

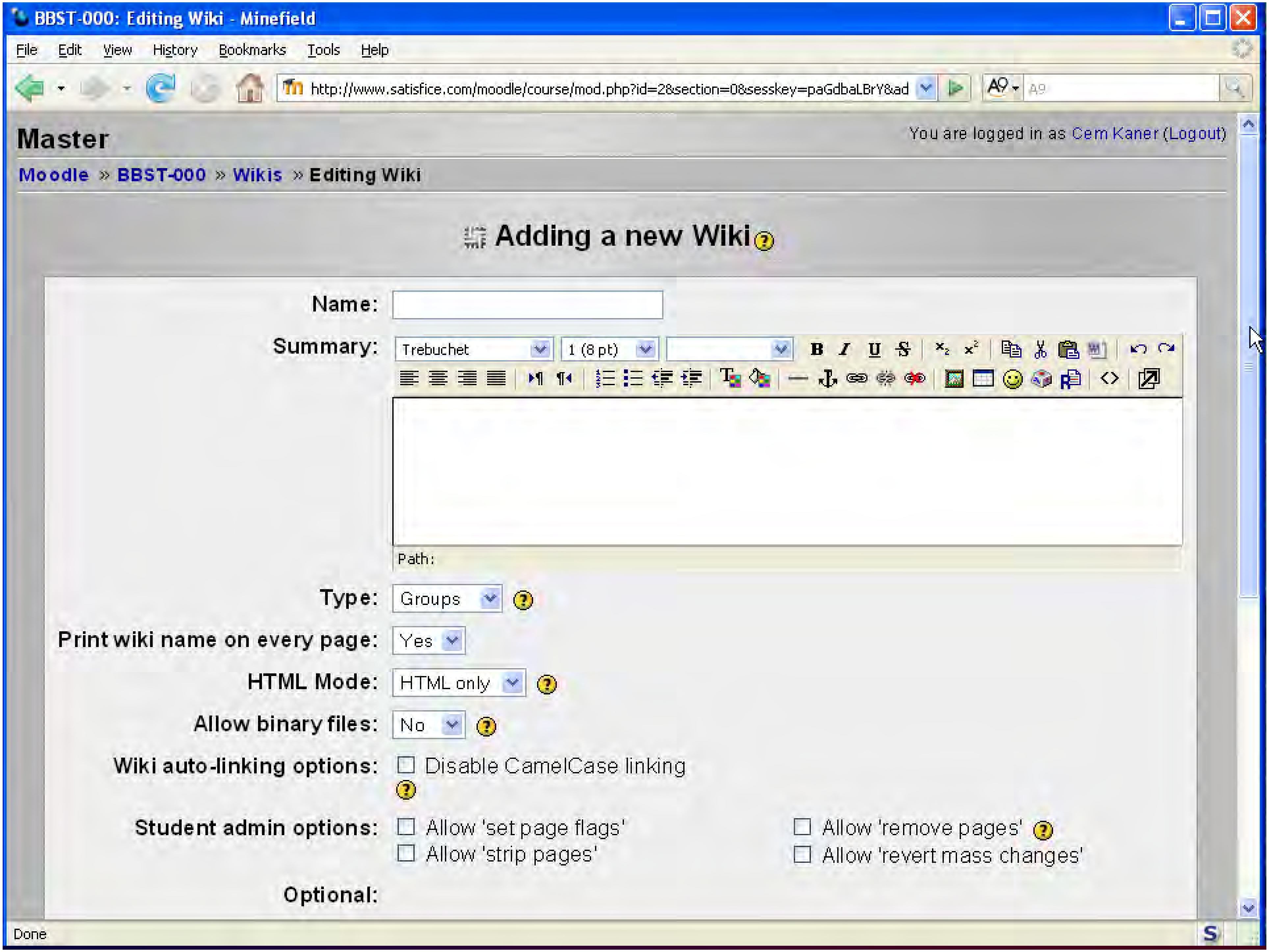
Upcoming Events

There are no upcoming events

Go to calendar...
New Event...

Recent Activity

Blocks


BBST-000: Editing Wiki - Minefield
File
Edit
View
History
Bookmarks
Tools
Help
http://www.satisfice.com/moodle/course/mod.php?id=2&section=0&sesskey=paGdbaLBrY&ad
Done

Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000 » Wikis » Editing Wiki

Adding a new Wiki

Name:

Summary: Trebuchet | 1 (8 pt)

Path:

Type: Groups

Print wiki name on every page: Yes

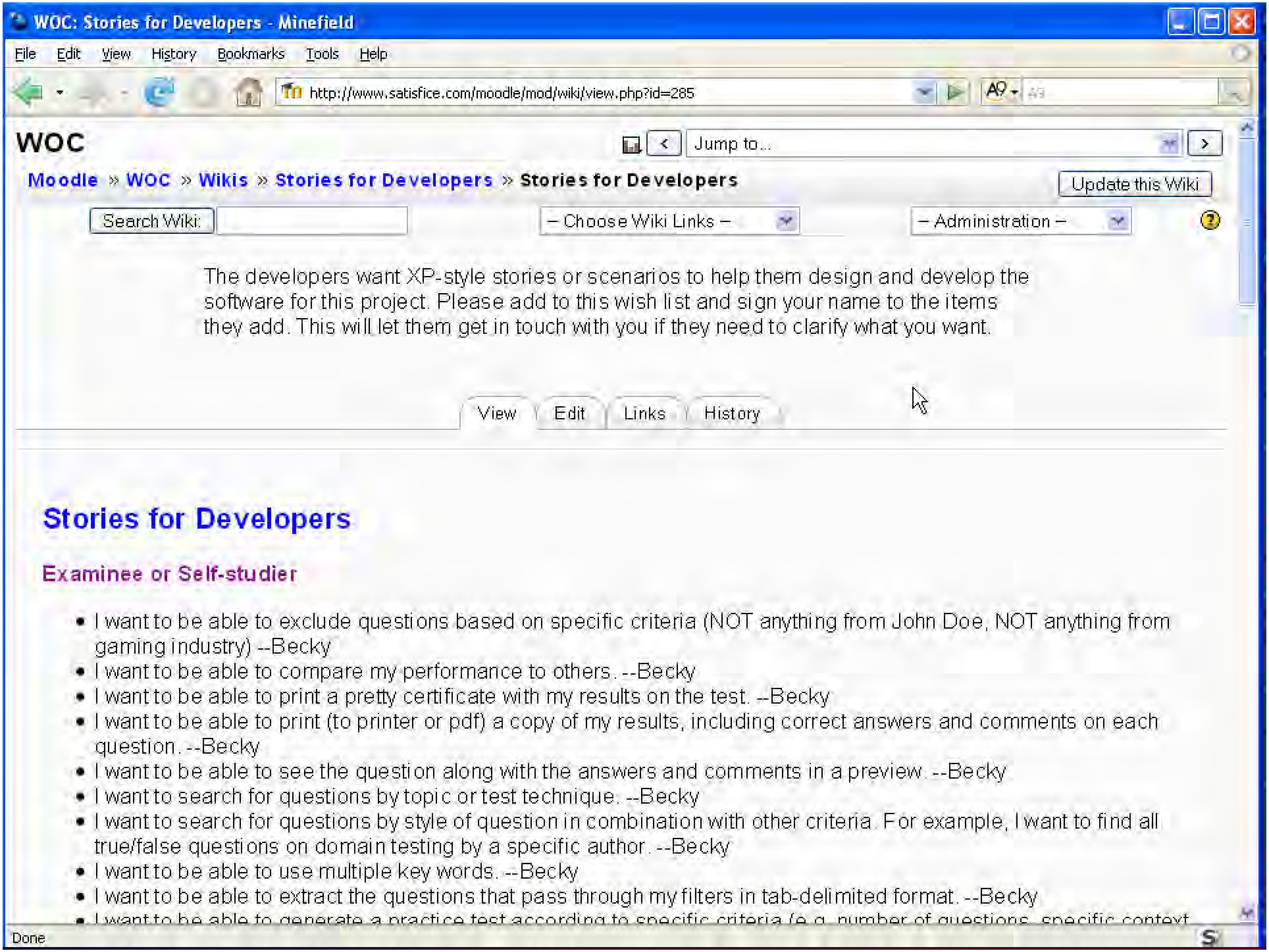
HTML Mode: HTML only

Allow binary files: No

Wiki auto-linking options: □ Disable CamelCase linking

Student admin options: □ Allow 'set page flags' □ Allow 'remove pages' □ Allow 'strip pages' □ Allow 'revert mass changes'

Optional:

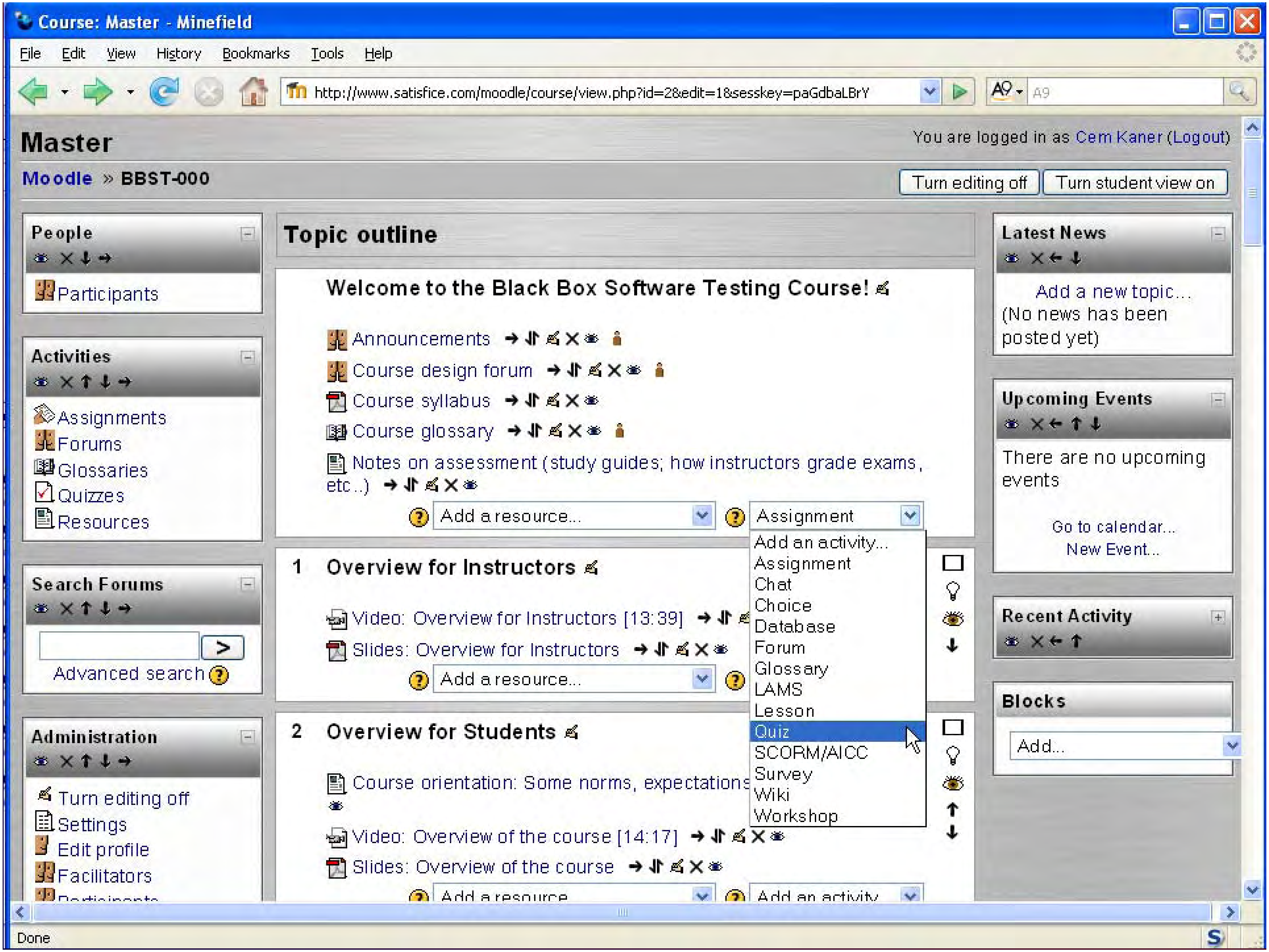

WOC: Stories for Developers - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/mod/wiki/view.php?id=285
WOC
Jump to...
Moodle » WOC » Wikis » Stories for Developers » Stories for Developers
Update this Wiki
Search Wiki:
– Choose Wiki Links –
– Administration –
View
Edit
Links
History
Done

The developers want XP-style stories or scenarios to help them design and develop the software for this project. Please add to this wish list and sign your name to the items they add. This will let them get in touch with you if they need to clarify what you want.

Stories for Developers

Examinee or Self-studier

- I want to be able to exclude questions based on specific criteria (NOT anything from John Doe, NOT anything from gaming industry) --Becky
- I want to be able to compare my performance to others. --Becky
- I want to be able to print a pretty certificate with my results on the test. --Becky
- I want to be able to print (to printer or pdf) a copy of my results, including correct answers and comments on each question. --Becky
- I want to be able to see the question along with the answers and comments in a preview. --Becky
- I want to search for questions by topic or test technique. --Becky
- I want to search for questions by style of question in combination with other criteria. For example, I want to find all true/false questions on domain testing by a specific author. --Becky
- I want to be able to use multiple key words. --Becky
- I want to be able to extract the questions that pass through my filters in tab-delimited format. --Becky
- I want to be able to generate a practice test according to specific criteria (e.g. number of questions, specific context



Course: Master - Minefield

File Edit View History Bookmarks Tools Help

http://www.satisfice.com/moodle/course/view.php?id=2&edit=1&sesskey=paGdbaLBrY

Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000

Turn editing off | Turn student view on

People

Participants

Activities

Assignments
Forums
Glossaries
Quizzes
Resources

Search Forums

>

Advanced search

Administration

Turn editing off
Settings
Edit profile
Facilitators
Participants

Topic outline

Welcome to the Black Box Software Testing Course!

Announcements
Course design forum
Course syllabus
Course glossary
Notes on assessment (study guides; how instructors grade exams, etc..)

Add a resource... | Assignment

Add an activity...
Assignment
Chat
Choice
Database
Forum
Glossary
LAMS
Lesson
Quiz
SCORM/AICC
Survey
Wiki
Workshop

1 Overview for Instructors

Video: Overview for Instructors [13:39]
Slides: Overview for Instructors

Add a resource...

2 Overview for Students

Course orientation: Some norms, expectations
Video: Overview of the course [14:17]
Slides: Overview of the course

Add a resource... | Add an activity...

Latest News

Add a new topic...
(No news has been posted yet)

Upcoming Events

There are no upcoming events

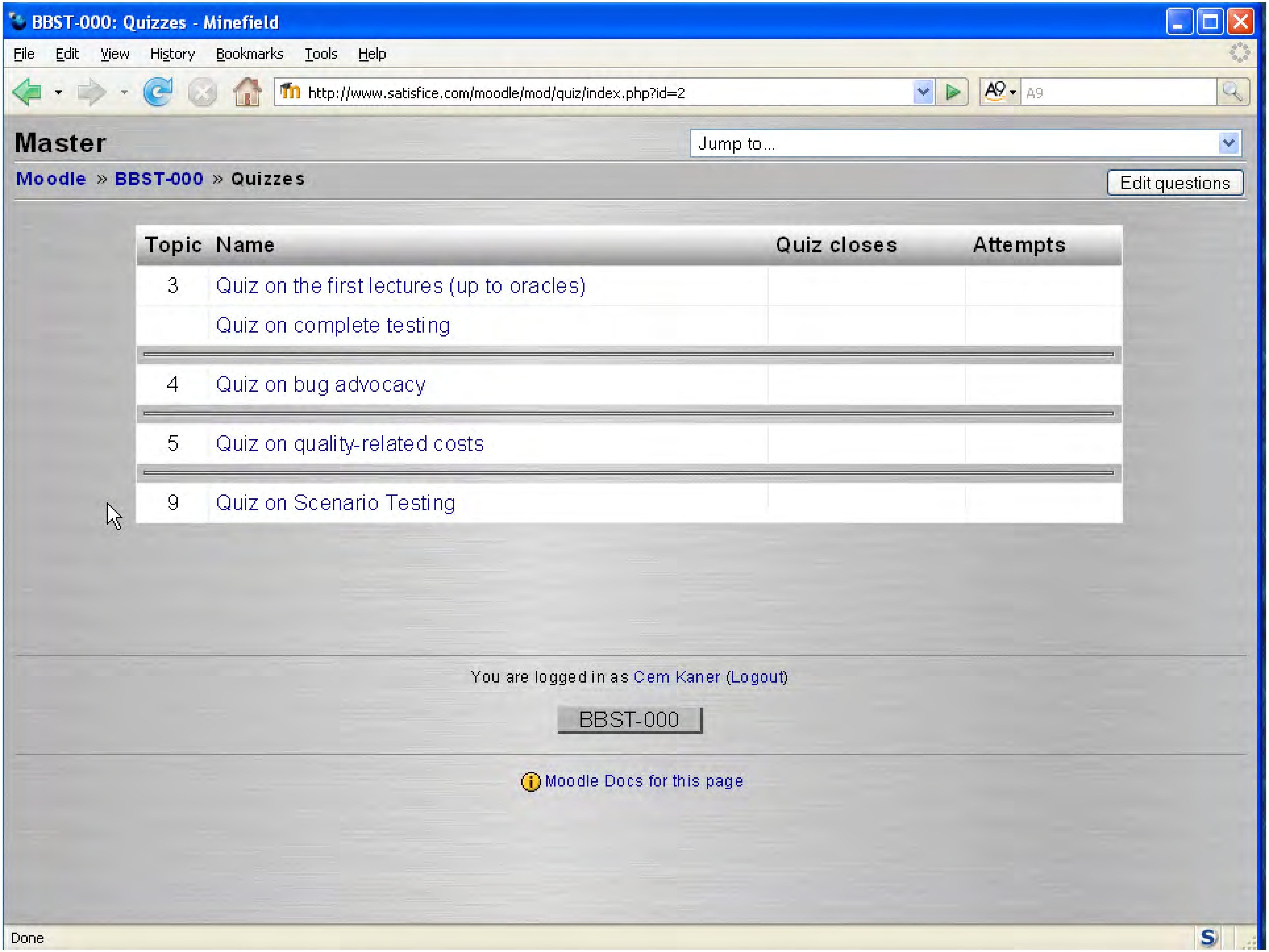
Go to calendar...
New Event...

Recent Activity

Blocks

Add...

Done



BBST-000: Quizzes - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/mod/quiz/index.php?id=2
Jump to...
Edit questions
Done

Master

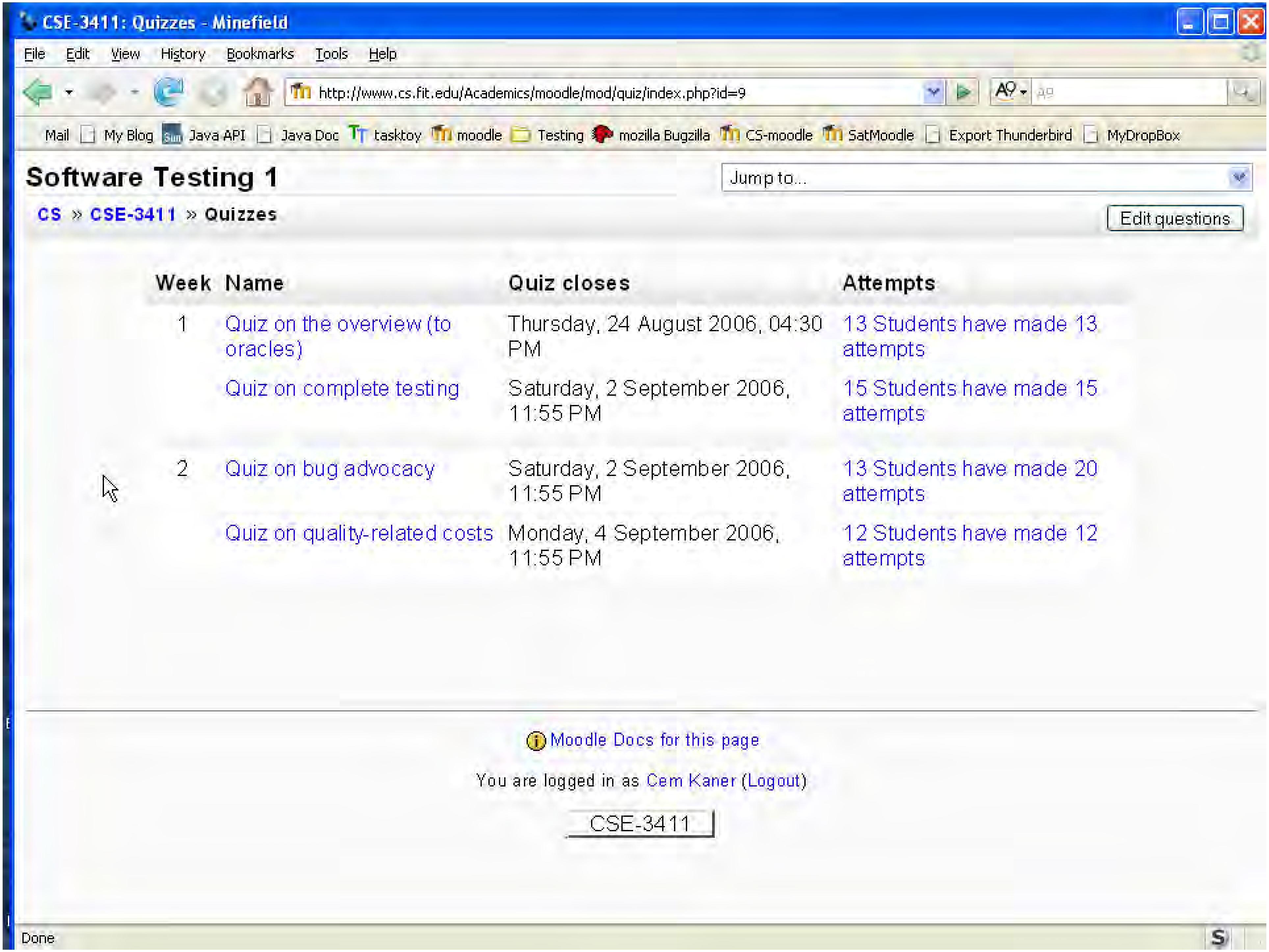
Moodle » BBST-000 » Quizzes

Topic	Name	Quiz closes	Attempts
3	Quiz on the first lectures (up to oracles)		
	Quiz on complete testing		
4	Quiz on bug advocacy		
5	Quiz on quality-related costs		
9	Quiz on Scenario Testing		

You are logged in as Cem Kaner (Logout)

BBST-000

Moodle Docs for this page



CSE-3411: Quizzes - Minefield
File Edit View History Bookmarks Tools Help
http://www.cs.fit.edu/Academics/moodle/mod/quiz/index.php?id=9
Mail My Blog Java API Java Doc tasktoy moodle Testing mozilla Bugzilla CS-moodle SatMoodle Export Thunderbird MyDropBox
Jump to...
Edit questions
Done

Software Testing 1

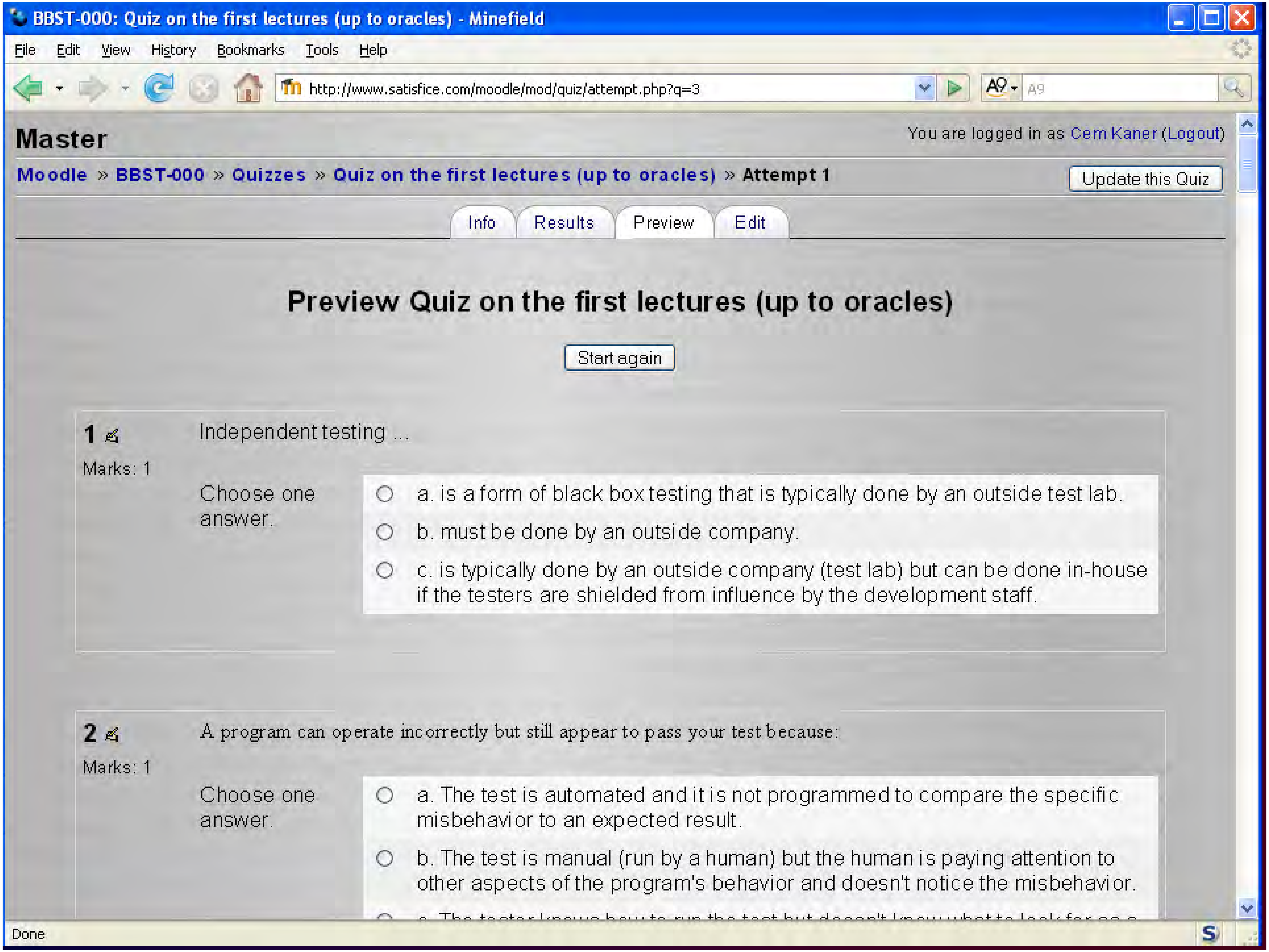
CS » CSE-3411 » Quizzes

Week	Name	Quiz closes	Attempts
1	Quiz on the overview (to oracles)	Thursday, 24 August 2006, 04:30 PM	13 Students have made 13 attempts
	Quiz on complete testing	Saturday, 2 September 2006, 11:55 PM	15 Students have made 15 attempts
2	Quiz on bug advocacy	Saturday, 2 September 2006, 11:55 PM	13 Students have made 20 attempts
	Quiz on quality-related costs	Monday, 4 September 2006, 11:55 PM	12 Students have made 12 attempts

Moodle Docs for this page

You are logged in as Cem Kaner (Logout)

CSE-3411


BBST-000: Quiz on the first lectures (up to oracles) - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/mod/quiz/attempt.php?q=3
Master
You are logged in as Cem Kaner (Logout)
Moodle » BBST-000 » Quizzes » Quiz on the first lectures (up to oracles) » Attempt 1
Update this Quiz
Info
Results
Preview
Edit
Start again
Done

Preview Quiz on the first lectures (up to oracles)

1
Marks: 1

Independent testing ...

Choose one answer.

○ a. is a form of black box testing that is typically done by an outside test lab.

○ b. must be done by an outside company.

○ c. is typically done by an outside company (test lab) but can be done in-house if the testers are shielded from influence by the development staff.

2
Marks: 1

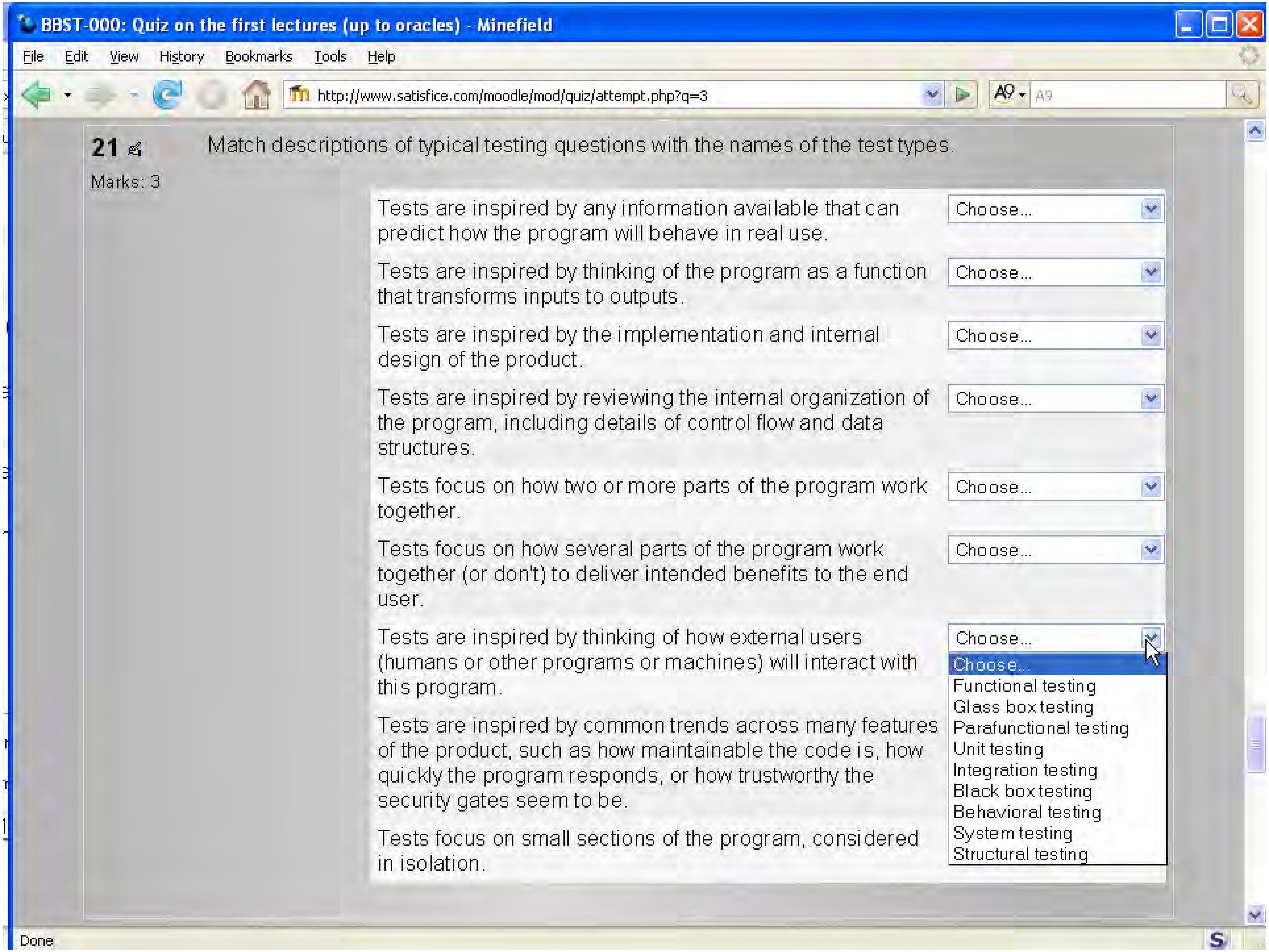
A program can operate incorrectly but still appear to pass your test because:

Choose one answer.

○ a. The test is automated and it is not programmed to compare the specific misbehavior to an expected result.

○ b. The test is manual (run by a human) but the human is paying attention to other aspects of the program's behavior and doesn't notice the misbehavior.

○ c. The tester knows how to run the test but doesn't know what to look for as a

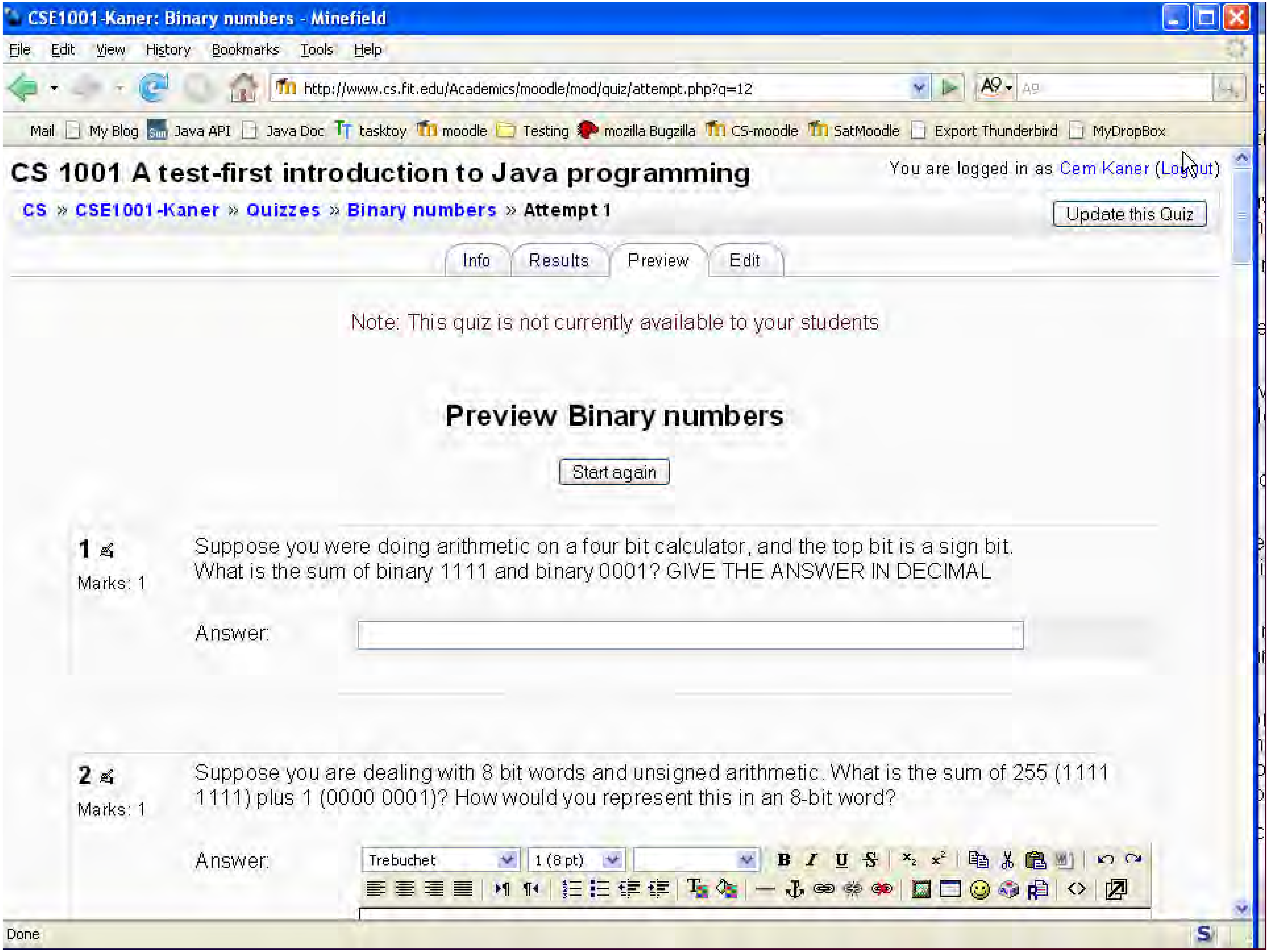

BBST-000: Quiz on the first lectures (up to oracles) - Minefield
File
Edit
View
History
Bookmarks
Tools
Help
http://www.satisfice.com/moodle/mod/quiz/attempt.php?q=3
A9
Done

21

Marks: 3

Match descriptions of typical testing questions with the names of the test types.

Tests are inspired by any information available that can predict how the program will behave in real use.	Choose...
Tests are inspired by thinking of the program as a function that transforms inputs to outputs.	Choose...
Tests are inspired by the implementation and internal design of the product.	Choose...
Tests are inspired by reviewing the internal organization of the program, including details of control flow and data structures.	Choose...
Tests focus on how two or more parts of the program work together.	Choose...
Tests focus on how several parts of the program work together (or don't) to deliver intended benefits to the end user.	Choose...
Tests are inspired by thinking of how external users (humans or other programs or machines) will interact with this program.	Choose... Choose... Functional testing Glass box testing Parafunctional testing Unit testing Integration testing Black box testing Behavioral testing System testing Structural testing
Tests are inspired by common trends across many features of the product, such as how maintainable the code is, how quickly the program responds, or how trustworthy the security gates seem to be.	
Tests focus on small sections of the program, considered in isolation.	


CSE1001-Kaner: Binary numbers - Minefield
File Edit View History Bookmarks Tools Help
http://www.cs.fit.edu/Academics/moodle/mod/quiz/attempt.php?q=12
Mail
My Blog
Java API
Java Doc
tasktoy
moodle
Testing
mozilla Bugzilla
CS-moodle
SatMoodle
Export Thunderbird
MyDropBox
You are logged in as Cem Kaner (Logout)
CS » CSE1001-Kaner » Quizzes » Binary numbers » Attempt 1
Update this Quiz
Info
Results
Preview
Edit
Start again
Trebuchet
1 (8 pt)
Done

CS 1001 A test-first introduction to Java programming

Note: This quiz is not currently available to your students

Preview Binary numbers

1
Marks: 1

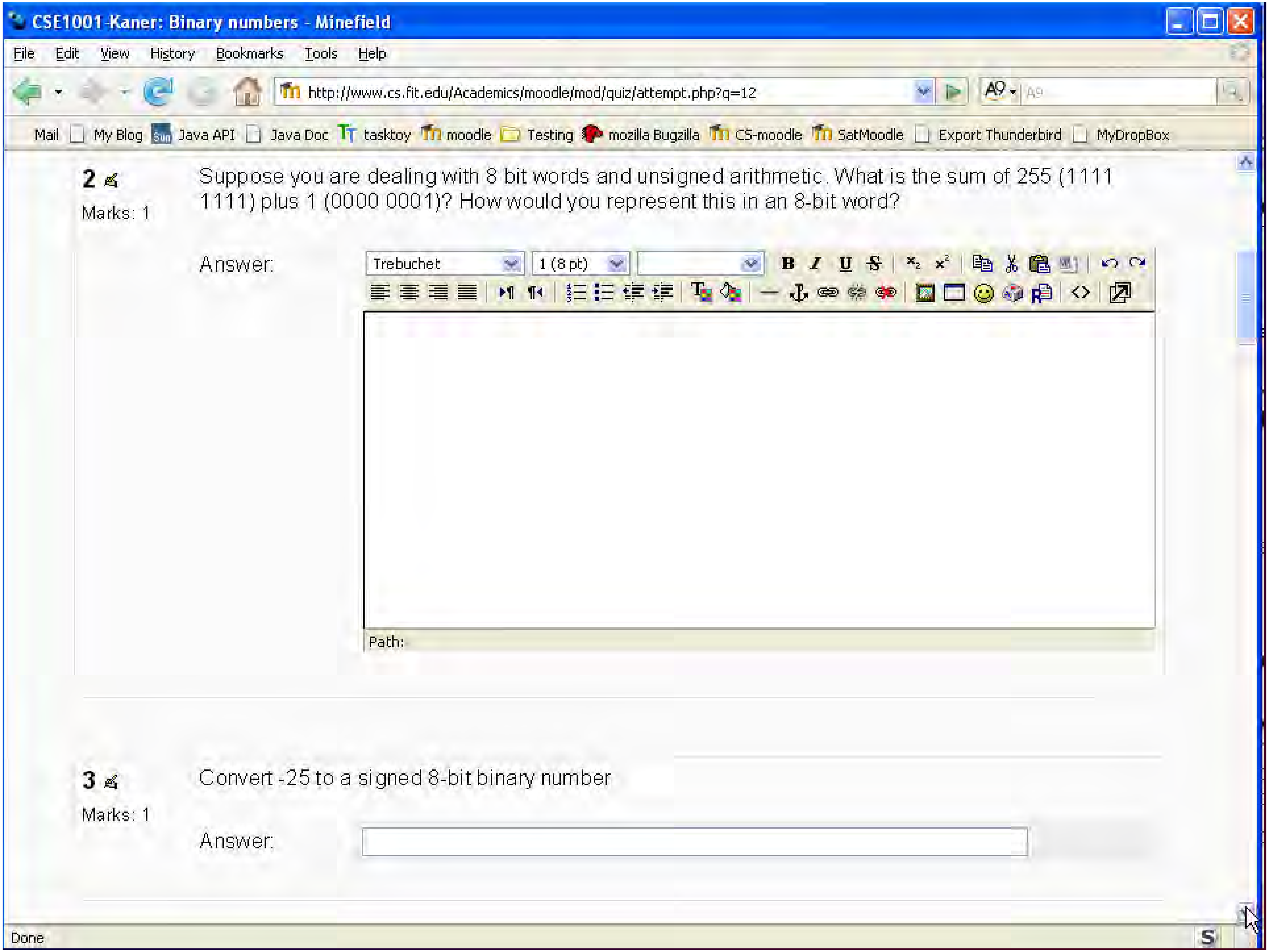
Suppose you were doing arithmetic on a four bit calculator, and the top bit is a sign bit. What is the sum of binary 1111 and binary 0001? GIVE THE ANSWER IN DECIMAL

Answer:

2
Marks: 1

Suppose you are dealing with 8 bit words and unsigned arithmetic. What is the sum of 255 (1111 1111) plus 1 (0000 0001)? How would you represent this in an 8-bit word?

Answer:


CSE1001-Kaner: Binary numbers - Minefield
File Edit View History Bookmarks Tools Help
http://www.cs.fit.edu/Academics/moodle/mod/quiz/attempt.php?q=12
Mail My Blog Java API Java Doc tasktoy moodle Testing mozilla Bugzilla CS-moodle SatMoodle Export Thunderbird MyDropBox

2

Marks: 1

Suppose you are dealing with 8 bit words and unsigned arithmetic. What is the sum of 255 (1111 1111) plus 1 (0000 0001)? How would you represent this in an 8-bit word?

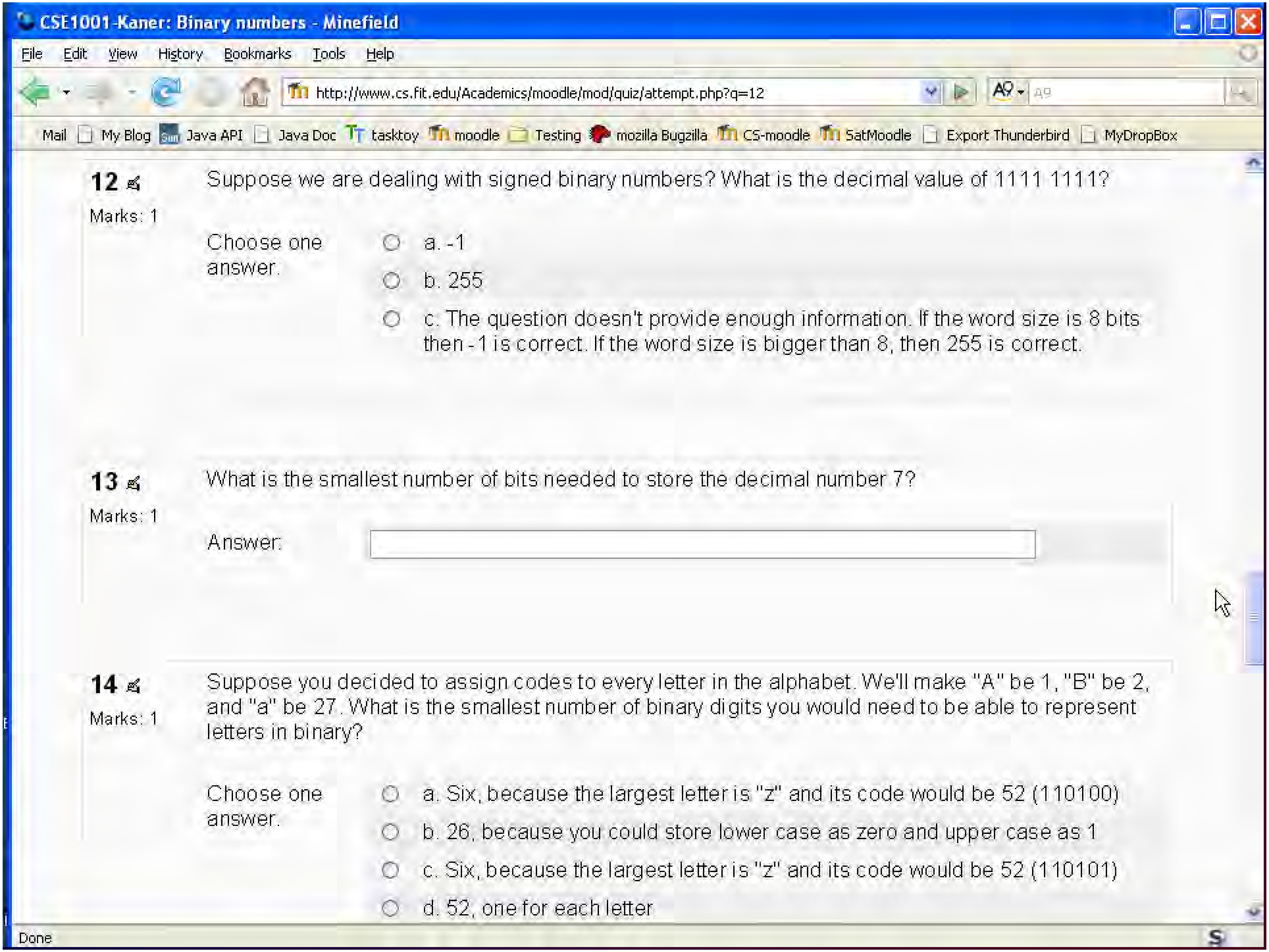
Answer:

3

Marks: 1

Convert -25 to a signed 8-bit binary number

Answer:


CSE1001-Kaner: Binary numbers - Minefield
File Edit View History Bookmarks Tools Help
http://www.cs.fit.edu/Academics/moodle/mod/quiz/attempt.php?q=12
Mail My Blog Java API Java Doc tasktoy moodle Testing mozilla Bugzilla CS-moodle SatMoodle Export Thunderbird MyDropBox
Done

12 Marks: 1

Suppose we are dealing with signed binary numbers? What is the decimal value of 1111 1111?

Choose one answer.

○ a. -1

○ b. 255

○ c. The question doesn't provide enough information. If the word size is 8 bits then -1 is correct. If the word size is bigger than 8, then 255 is correct.

13 Marks: 1

What is the smallest number of bits needed to store the decimal number 7?

Answer:

14 Marks: 1

Suppose you decided to assign codes to every letter in the alphabet. We'll make "A" be 1, "B" be 2, and "a" be 27. What is the smallest number of binary digits you would need to be able to represent letters in binary?

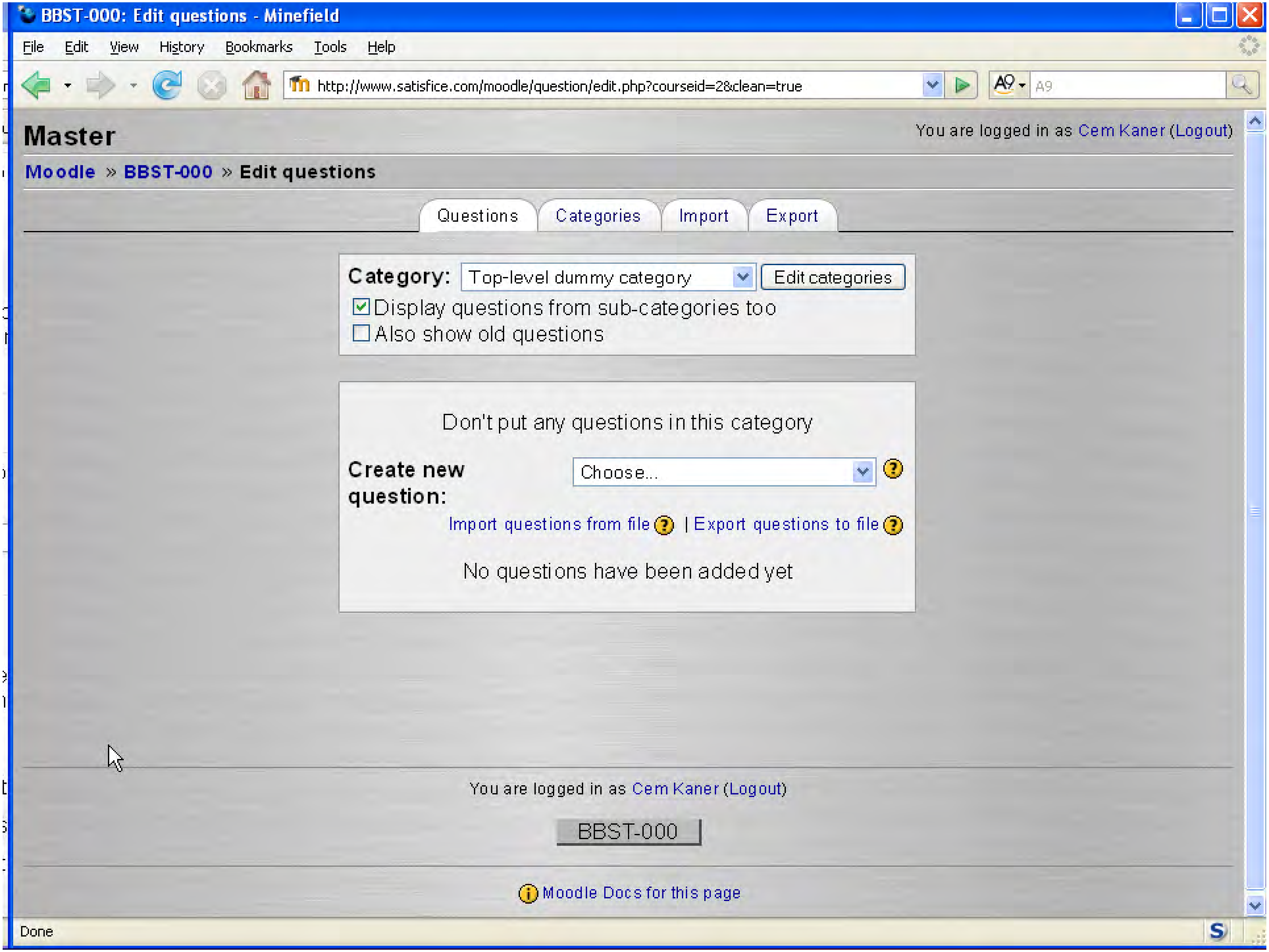
Choose one answer.

○ a. Six, because the largest letter is "z" and its code would be 52 (110100)

○ b. 26, because you could store lower case as zero and upper case as 1

○ c. Six, because the largest letter is "z" and its code would be 52 (110101)

○ d. 52, one for each letter



BBST-000: Edit questions - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/question/edit.php?courseid=2&clean=true

Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000 » Edit questions

Questions
Categories
Import
Export

Category: Top-level dummy category

☑ Display questions from sub-categories too

□ Also show old questions

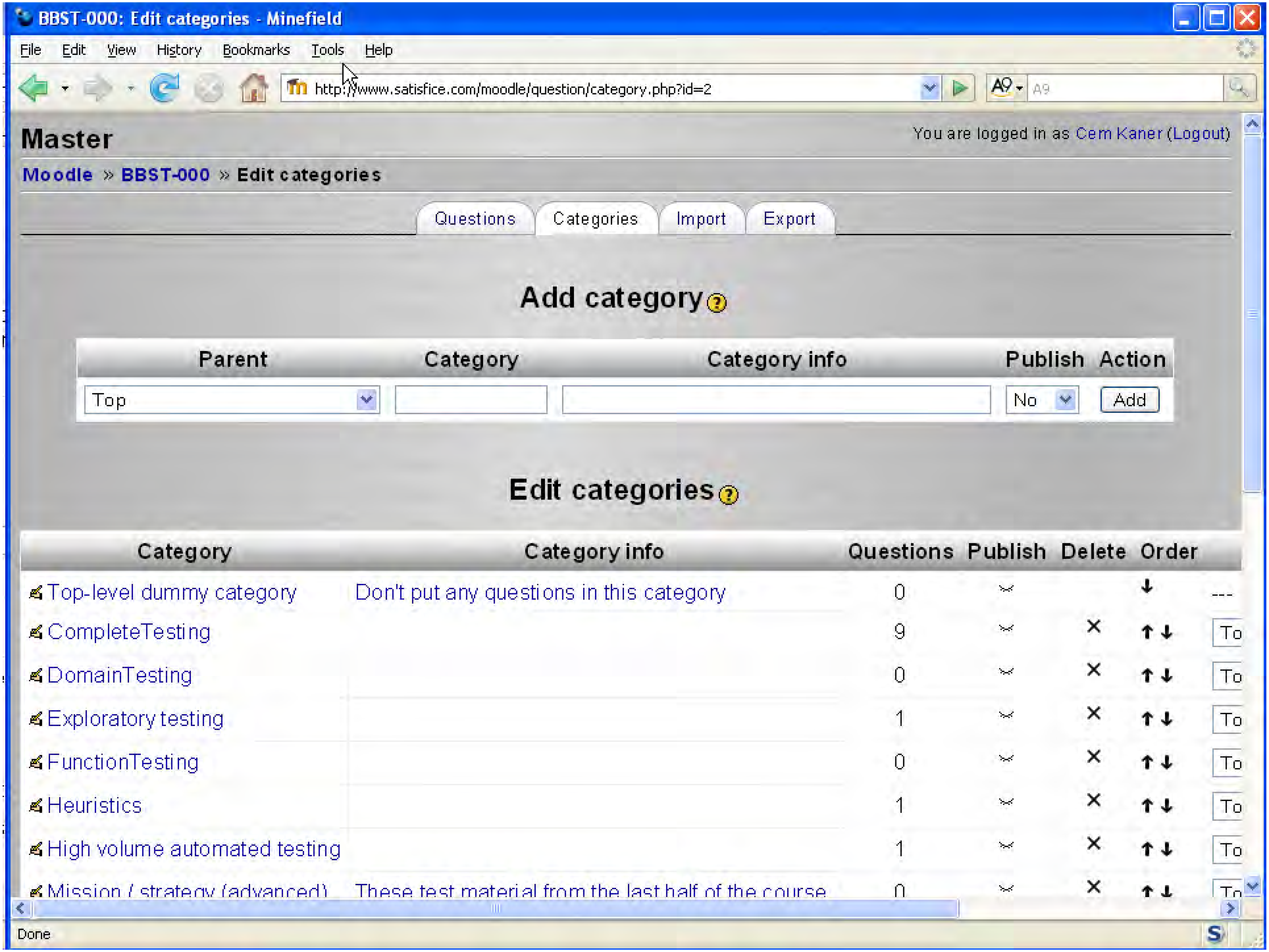
Don't put any questions in this category

Create new question: Choose...

Import questions from file | Export questions to file

No questions have been added yet

You are logged in as Cem Kaner (Logout)


BBST-000: Edit categories - Minefield
File
Edit
View
History
Bookmarks
Tools
Help
http://www.satisfice.com/moodle/question/category.php?id=2
A9
Done

Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000 » Edit categories

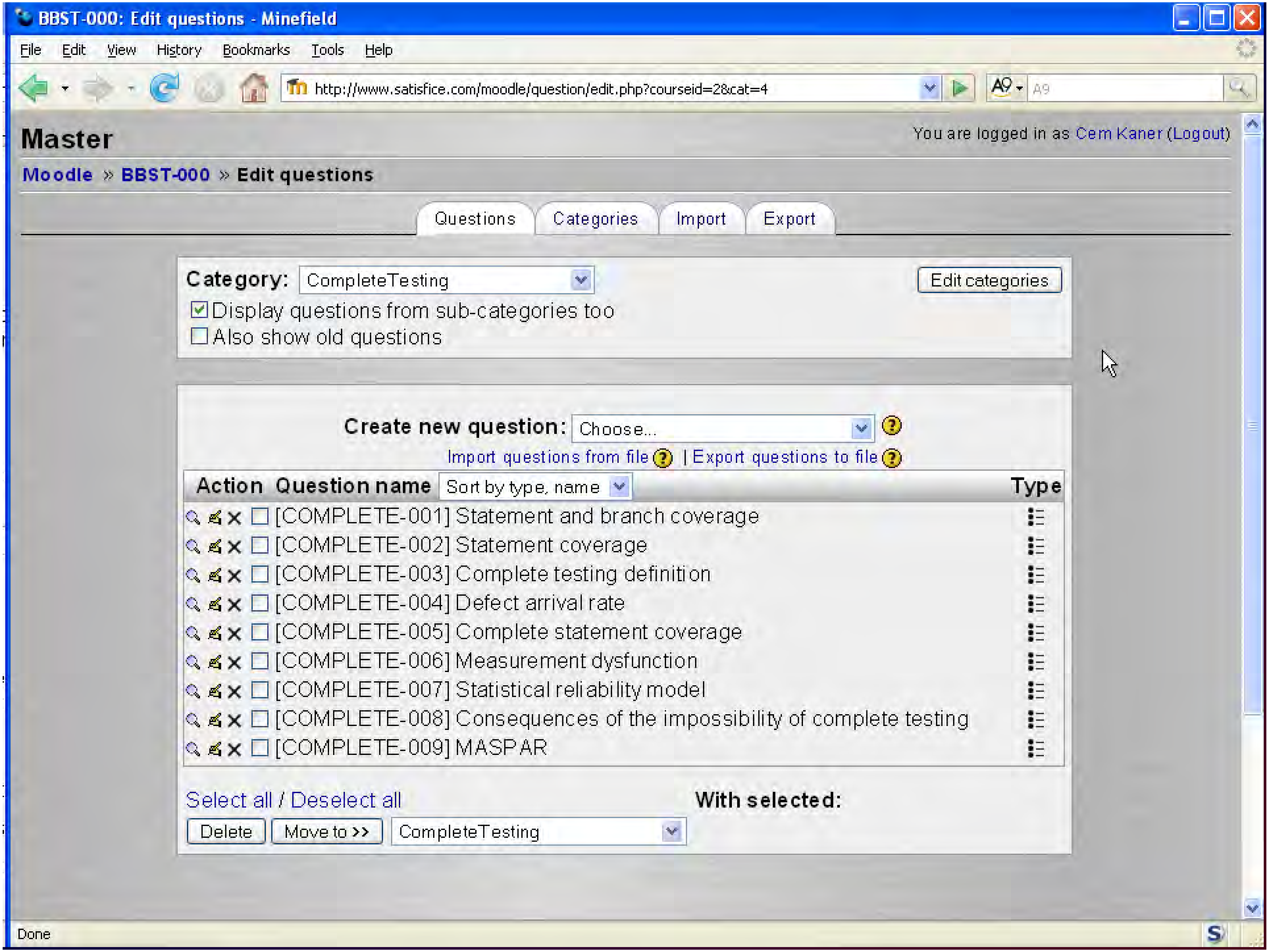
Questions
Categories
Import
Export

Add category

Parent	Category	Category info	Publish	Action
Top			No	Add

Edit categories

Category	Category info	Questions	Publish	Delete	Order	
Top-level dummy category	Don't put any questions in this category	0			↓	---
CompleteTesting		9		×	↑ ↓	To
DomainTesting		0		×	↑ ↓	To
Exploratory testing		1		×	↑ ↓	To
FunctionTesting		0		×	↑ ↓	To
Heuristics		1		×	↑ ↓	To
High volume automated testing		1		×	↑ ↓	To
Mission / strategy (advanced)	These test material from the last half of the course	0		×	↑ ↓	To



BBST-000: Edit questions - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/question/edit.php?courseid=2&cat=4

Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000 » Edit questions

Questions Categories Import Export

Category: CompleteTesting

☑ Display questions from sub-categories too

□ Also show old questions

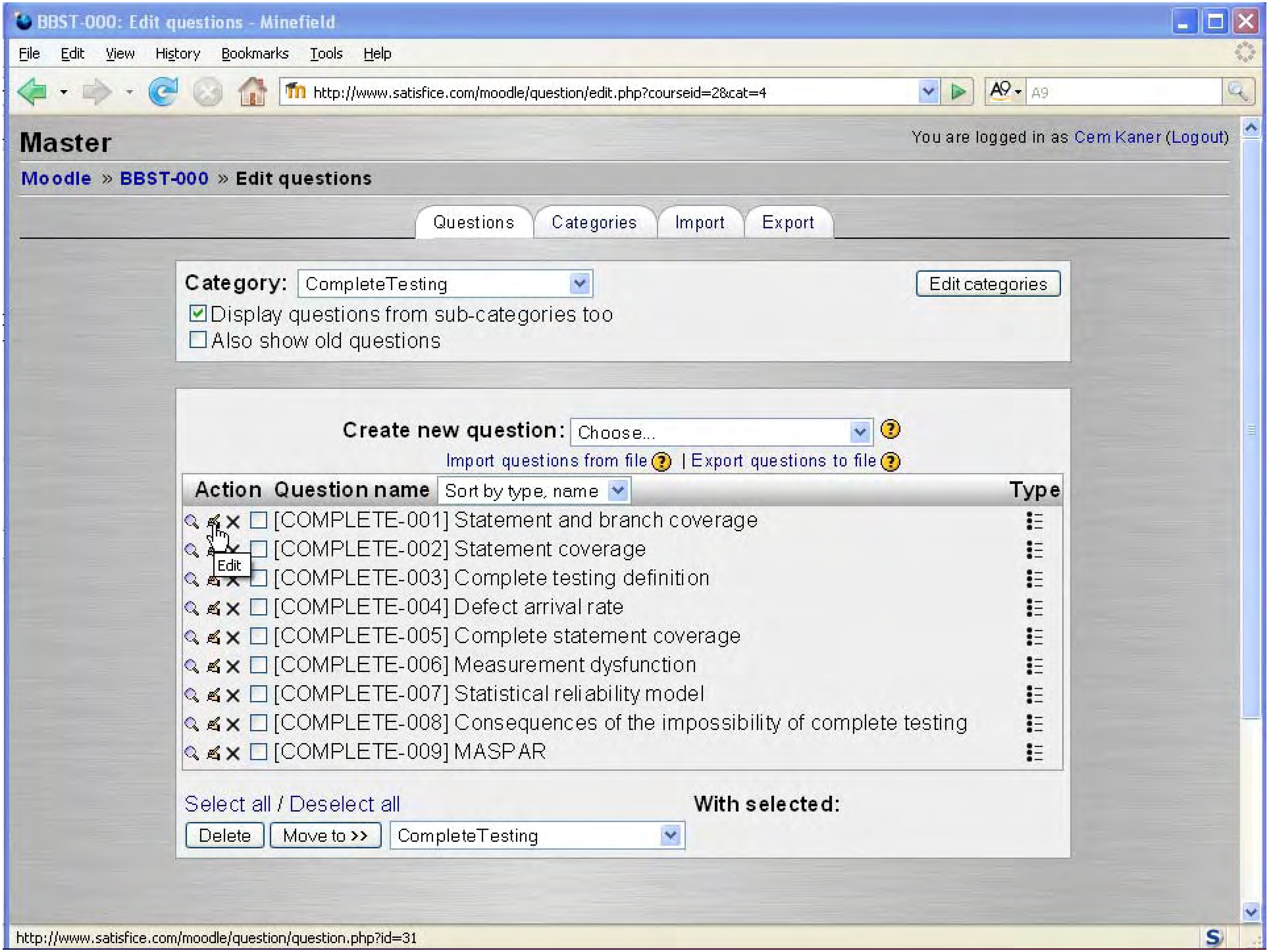
Create new question: Choose...

Import questions from file | Export questions to file

Action	Question name	Sort by type, name	Type
□	[COMPLETE-001] Statement and branch coverage		
□	[COMPLETE-002] Statement coverage		
□	[COMPLETE-003] Complete testing definition		
□	[COMPLETE-004] Defect arrival rate		
□	[COMPLETE-005] Complete statement coverage		
□	[COMPLETE-006] Measurement dysfunction		
□	[COMPLETE-007] Statistical reliability model		
□	[COMPLETE-008] Consequences of the impossibility of complete testing		
□	[COMPLETE-009] MASPAR		

Select all / Deselect all

With selected:



BBST-000: Edit questions - Minefield
File Edit View History Bookmarks Tools Help
A9
Questions
Categories
Import
Export
Edit categories
Delete
Move to >>
Edit

http://www.satisfice.com/moodle/question/edit.php?courseid=2&cat=4

Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000 » Edit questions

Category: CompleteTesting

☑ Display questions from sub-categories too

□ Also show old questions

Create new question: Choose...

Import questions from file | Export questions to file

Action Question name Sort by type, name Type

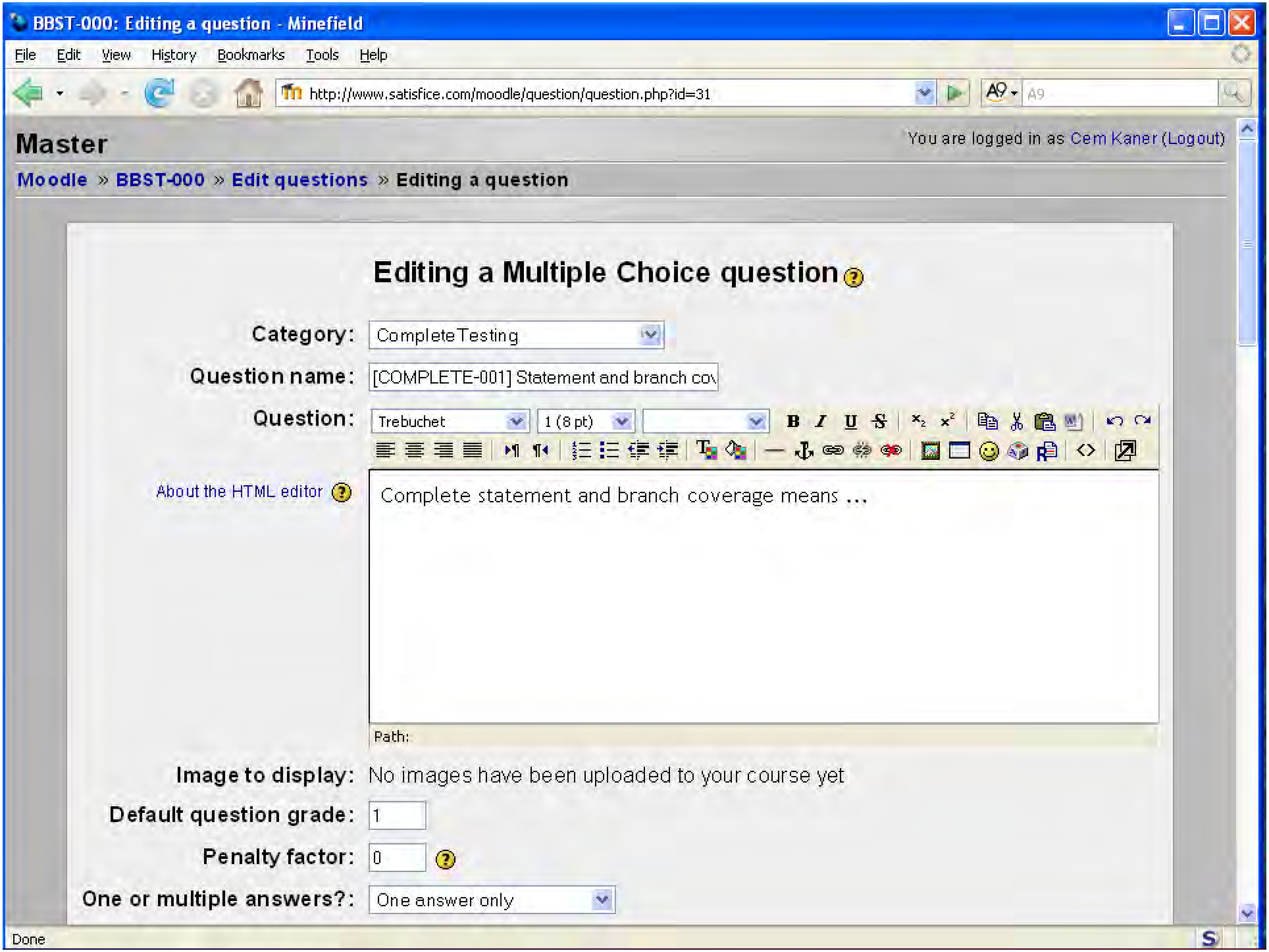
- □ [COMPLETE-001] Statement and branch coverage
- □ [COMPLETE-002] Statement coverage
- □ [COMPLETE-003] Complete testing definition
- □ [COMPLETE-004] Defect arrival rate
- □ [COMPLETE-005] Complete statement coverage
- □ [COMPLETE-006] Measurement dysfunction
- □ [COMPLETE-007] Statistical reliability model
- □ [COMPLETE-008] Consequences of the impossibility of complete testing
- □ [COMPLETE-009] MASPAR

Select all / Deselect all

With selected:

CompleteTesting

http://www.satisfice.com/moodle/question/question.php?id=31


BBST-000: Editing a question - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/question/question.php?id=31
Master
You are logged in as Cem Kaner (Logout)
Moodle » BBST-000 » Edit questions » Editing a question
About the HTML editor
Trebuchet
1 (8 pt)
Path:
Done

Editing a Multiple Choice question

Category: CompleteTesting

Question name: [COMPLETE-001] Statement and branch cov

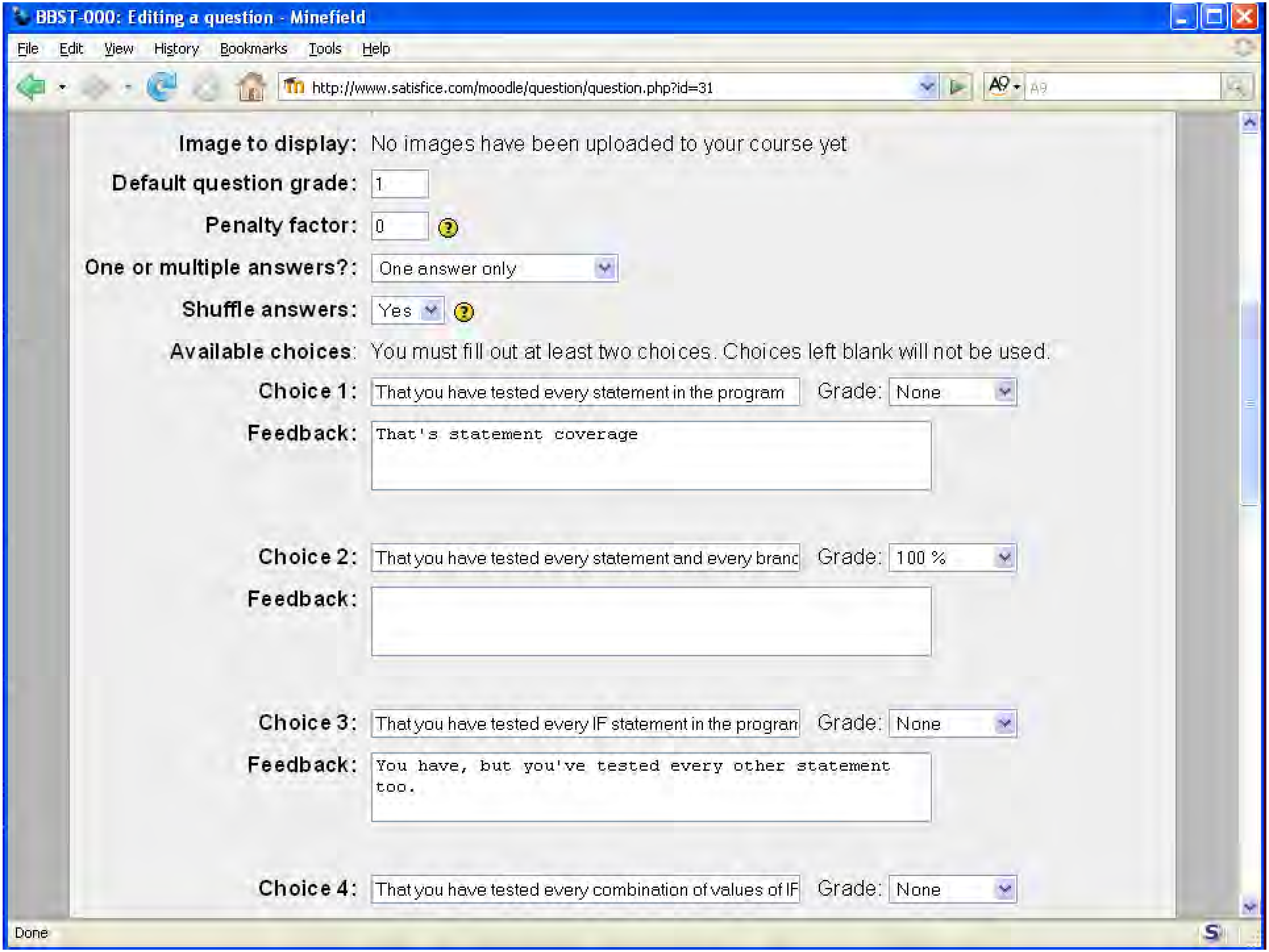
Question: Complete statement and branch coverage means …

Image to display: No images have been uploaded to your course yet

Default question grade: 1

Penalty factor: 0

One or multiple answers?: One answer only



BBST-000: Editing a question - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/question/question.php?id=31
Done

Image to display: No images have been uploaded to your course yet

Default question grade: 1

Penalty factor: 0

One or multiple answers?: One answer only

Shuffle answers: Yes

Available choices: You must fill out at least two choices. Choices left blank will not be used.

Choice 1: That you have tested every statement in the program Grade: None

Feedback: That's statement coverage

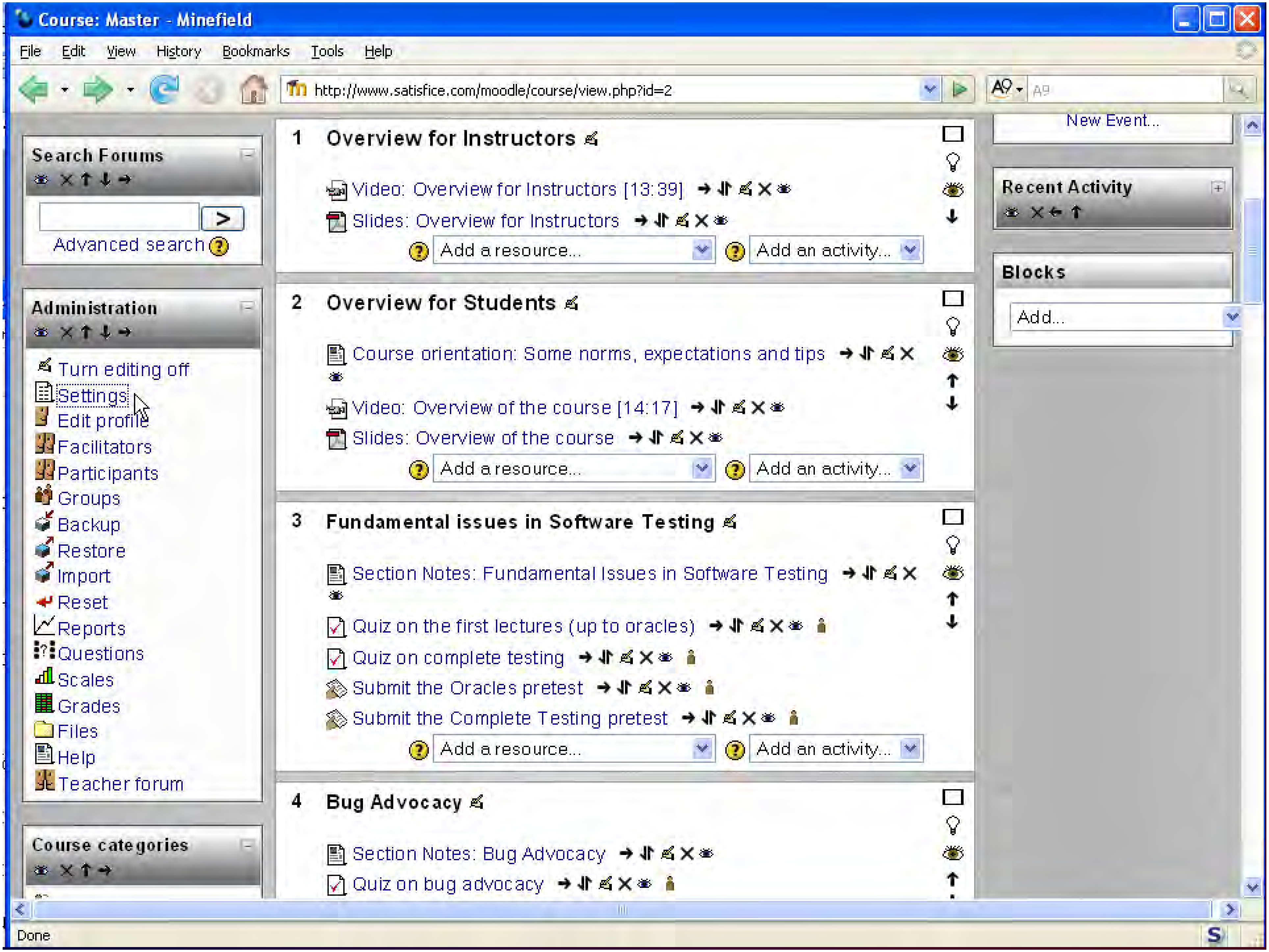
Choice 2: That you have tested every statement and every branc Grade: 100 %

Feedback:

Choice 3: That you have tested every IF statement in the progran Grade: None

Feedback: You have, but you've tested every other statement too.

Choice 4: That you have tested every combination of values of IF Grade: None


Course: Master - Minefield
File
Edit
View
History
Bookmarks
Tools
Help
http://www.satisfice.com/moodle/course/view.php?id=2
Done

Search Forums

>

Advanced search

Administration

- Turn editing off
- Settings
- Edit profile
- Facilitators
- Participants
- Groups
- Backup
- Restore
- Import
- Reset
- Reports
- Questions
- Scales
- Grades
- Files
- Help
- Teacher forum

Course categories

1 **Overview for Instructors**

- Video: Overview for Instructors [13:39]
- Slides: Overview for Instructors

Add a resource... Add an activity...

2 **Overview for Students**

- Course orientation: Some norms, expectations and tips
- Video: Overview of the course [14:17]
- Slides: Overview of the course

Add a resource... Add an activity...

3 **Fundamental issues in Software Testing**

- Section Notes: Fundamental Issues in Software Testing
- Quiz on the first lectures (up to oracles)
- Quiz on complete testing
- Submit the Oracles pretest
- Submit the Complete Testing pretest

Add a resource... Add an activity...

4 **Bug Advocacy**

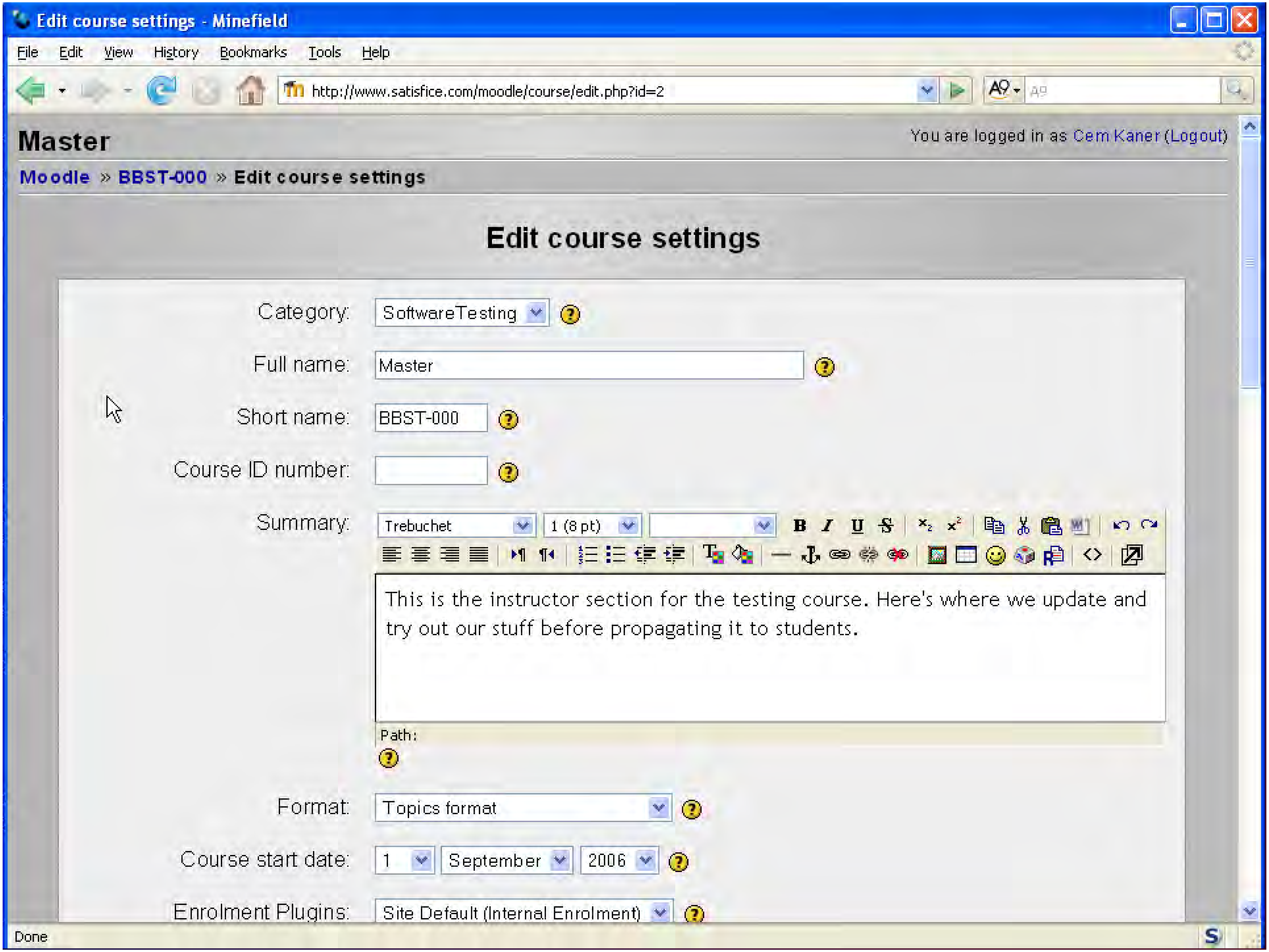
- Section Notes: Bug Advocacy
- Quiz on bug advocacy

New Event...

Recent Activity

Blocks

Add...


Edit course settings - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/course/edit.php?id=2
Done

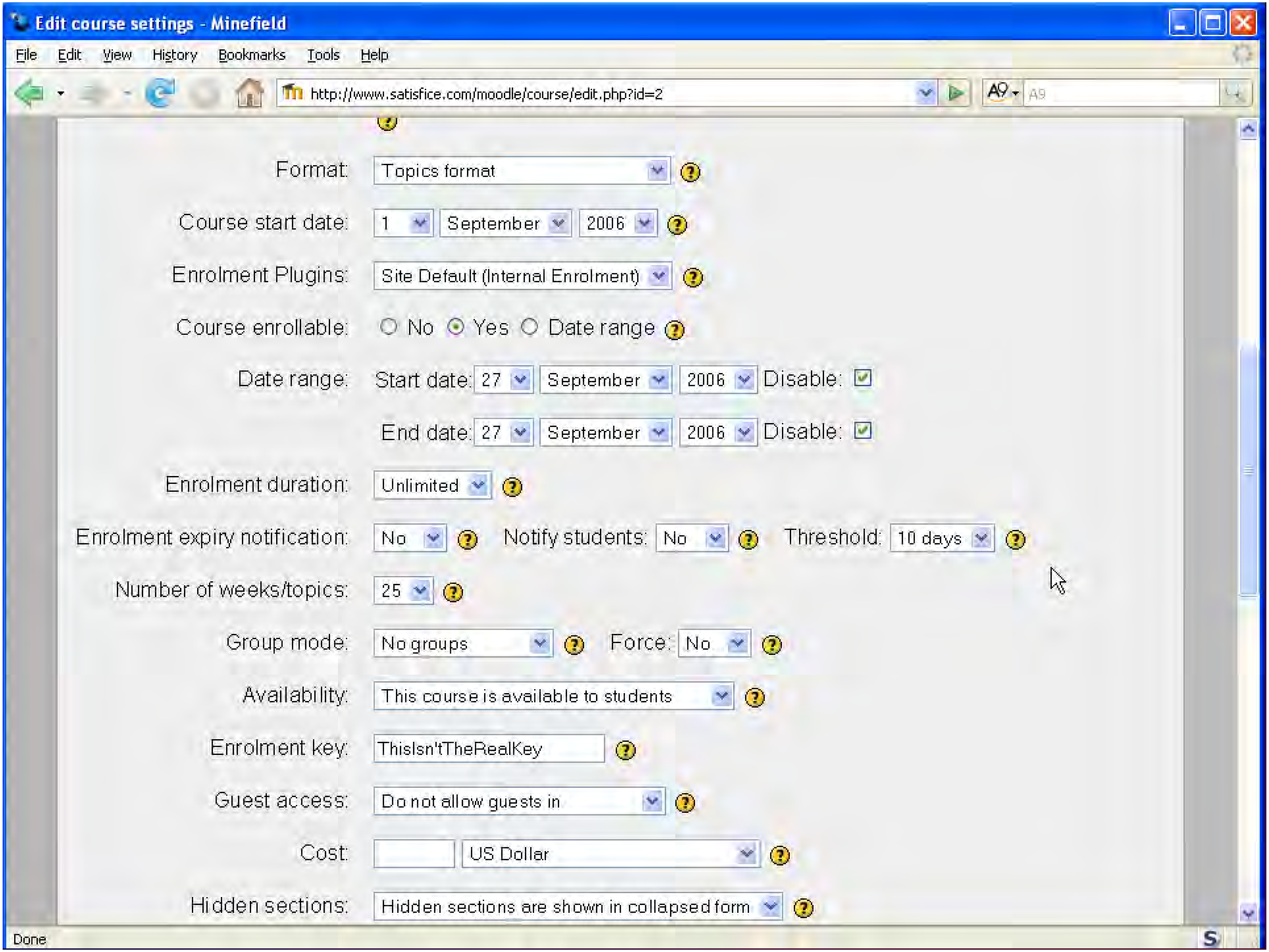
Master

You are logged in as Cem Kaner (Logout)

Moodle » BBST-000 » Edit course settings

Edit course settings

Category:	SoftwareTesting
Full name:	Master
Short name:	BBST-000
Course ID number:	
Summary:	This is the instructor section for the testing course. Here's where we update and try out our stuff before propagating it to students.
Format:	Topics format
Course start date:	1 September 2006
Enrolment Plugins:	Site Default (Internal Enrolment)


Edit course settings - Minefield
File
Edit
View
History
Bookmarks
Tools
Help
http://www.satisfice.com/moodle/course/edit.php?id=2
Done

Format: Topics format

Course start date: 1 September 2006

Enrolment Plugins: Site Default (Internal Enrolment)

Course enrollable: ○ No ● Yes ○ Date range

Date range: Start date: 27 September 2006 Disable: ☑

End date: 27 September 2006 Disable: ☑

Enrolment duration: Unlimited

Enrolment expiry notification: No Notify students: No Threshold: 10 days

Number of weeks/topics: 25

Group mode: No groups Force: No

Availability: This course is available to students

Enrolment key: ThisIsn'tTheRealKey

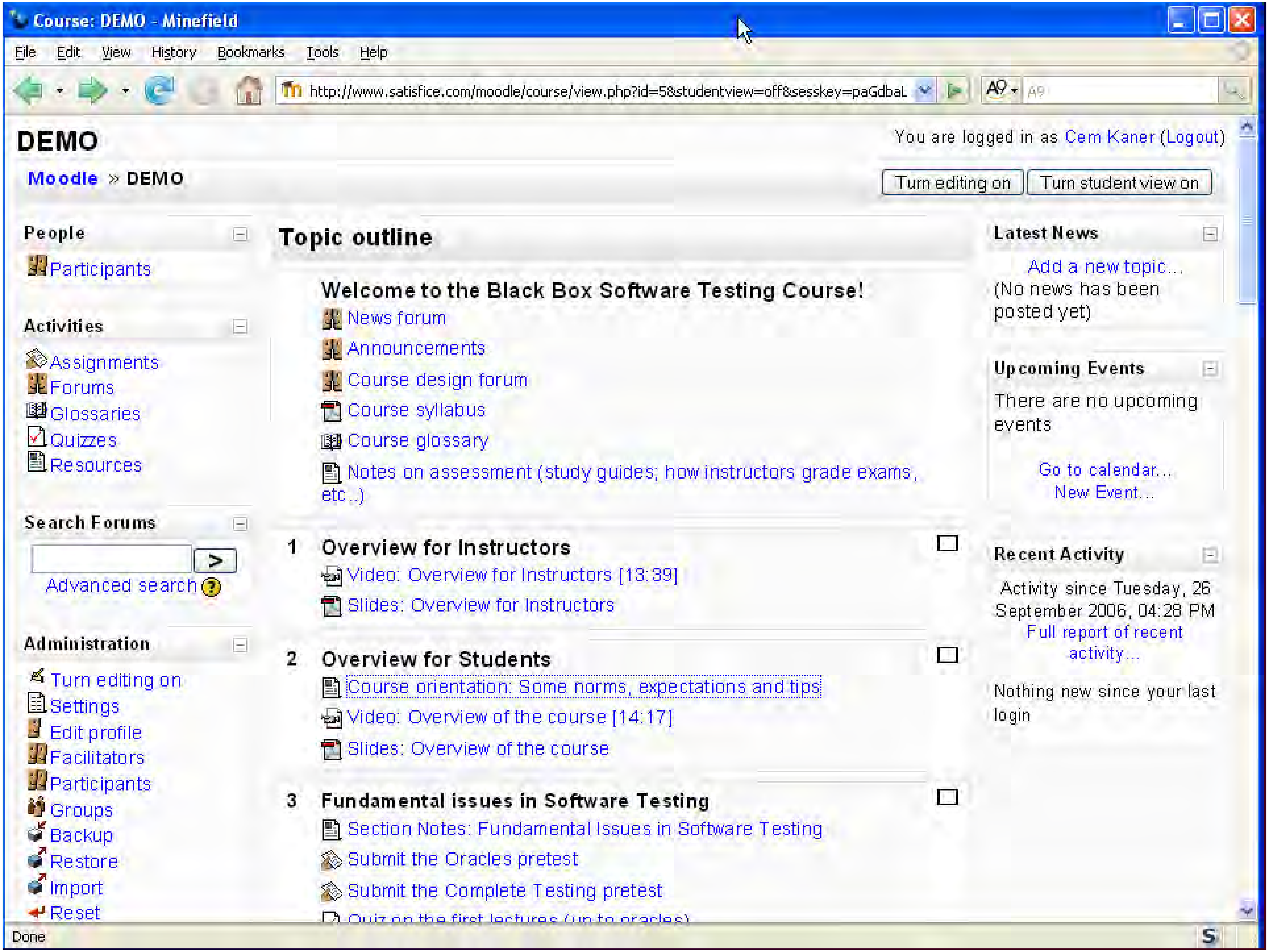
Guest access: Do not allow guests in

Cost: US Dollar

Hidden sections: Hidden sections are shown in collapsed form

Overview

1. *Tour of the Moodle course management system*
2. *Tour of the Black Box Software Testing Course on Moodle*
3. *Overview of use of material like this in the workplace*
4. *Dealing with the instructional challenges of a cognitively complex field of study*
5. *Taking control of your learning objectives for the course*
6. *Examples of activity patterns*


Course: DEMO - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/course/view.php?id=5&studentview=off&sesskey=paGdbaL
A9
Turn editing on
Turn student view on
>
Done

DEMO

You are logged in as Cem Kaner (Logout)

Moodle » DEMO

People

- Participants

Activities

- Assignments
- Forums
- Glossaries
- Quizzes
- Resources

Search Forums

Advanced search

Administration

- Turn editing on
- Settings
- Edit profile
- Facilitators
- Participants
- Groups
- Backup
- Restore
- Import
- Reset

Topic outline

Welcome to the Black Box Software Testing Course!

- News forum
- Announcements
- Course design forum
- Course syllabus
- Course glossary
- Notes on assessment (study guides; how instructors grade exams, etc.)

1 **Overview for Instructors** □

- Video: Overview for Instructors [13:39]
- Slides: Overview for Instructors

2 **Overview for Students** □

- Course orientation: Some norms, expectations and tips
- Video: Overview of the course [14:17]
- Slides: Overview of the course

3 **Fundamental issues in Software Testing** □

- Section Notes: Fundamental Issues in Software Testing
- Submit the Oracles pretest
- Submit the Complete Testing pretest
- Quiz on the first lectures (up to oracles)

Latest News

Add a new topic...
(No news has been posted yet)

Upcoming Events

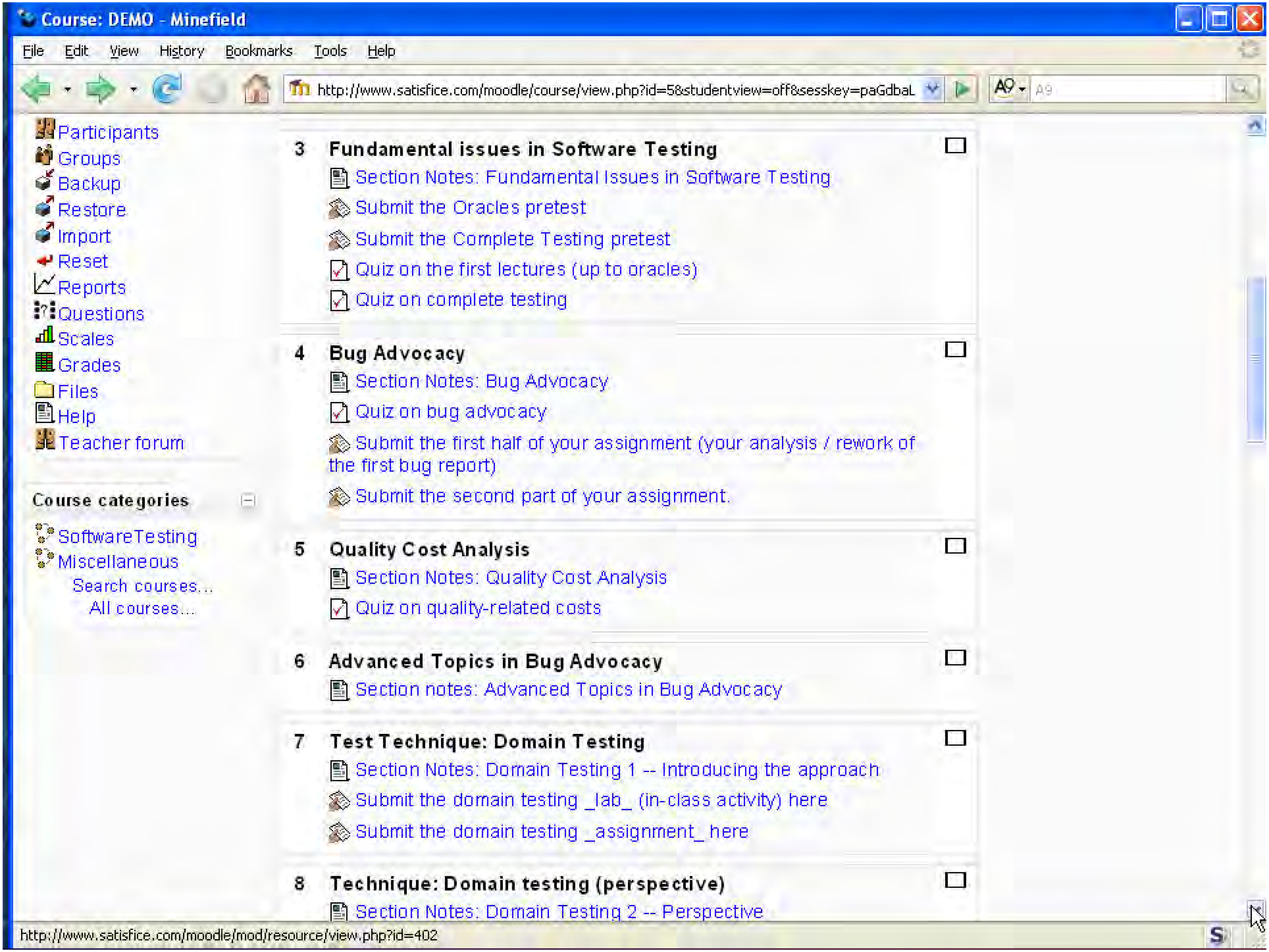
There are no upcoming events

Go to calendar...
New Event...

Recent Activity

Activity since Tuesday, 26 September 2006, 04:28 PM
Full report of recent activity...

Nothing new since your last login


Course: DEMO - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/course/view.php?id=5&studentview=off&sesskey=paGdbaL
A9

- Participants
- Groups
- Backup
- Restore
- Import
- Reset
- Reports
- Questions
- Scales
- Grades
- Files
- Help
- Teacher forum

Course categories

- SoftwareTesting
- Miscellaneous
- Search courses...
- All courses...

3 **Fundamental issues in Software Testing** □
- Section Notes: Fundamental Issues in Software Testing
- Submit the Oracles pretest
- Submit the Complete Testing pretest
- Quiz on the first lectures (up to oracles)
- Quiz on complete testing

4 **Bug Advocacy** □
- Section Notes: Bug Advocacy
- Quiz on bug advocacy
- Submit the first half of your assignment (your analysis / rework of the first bug report)
- Submit the second part of your assignment.

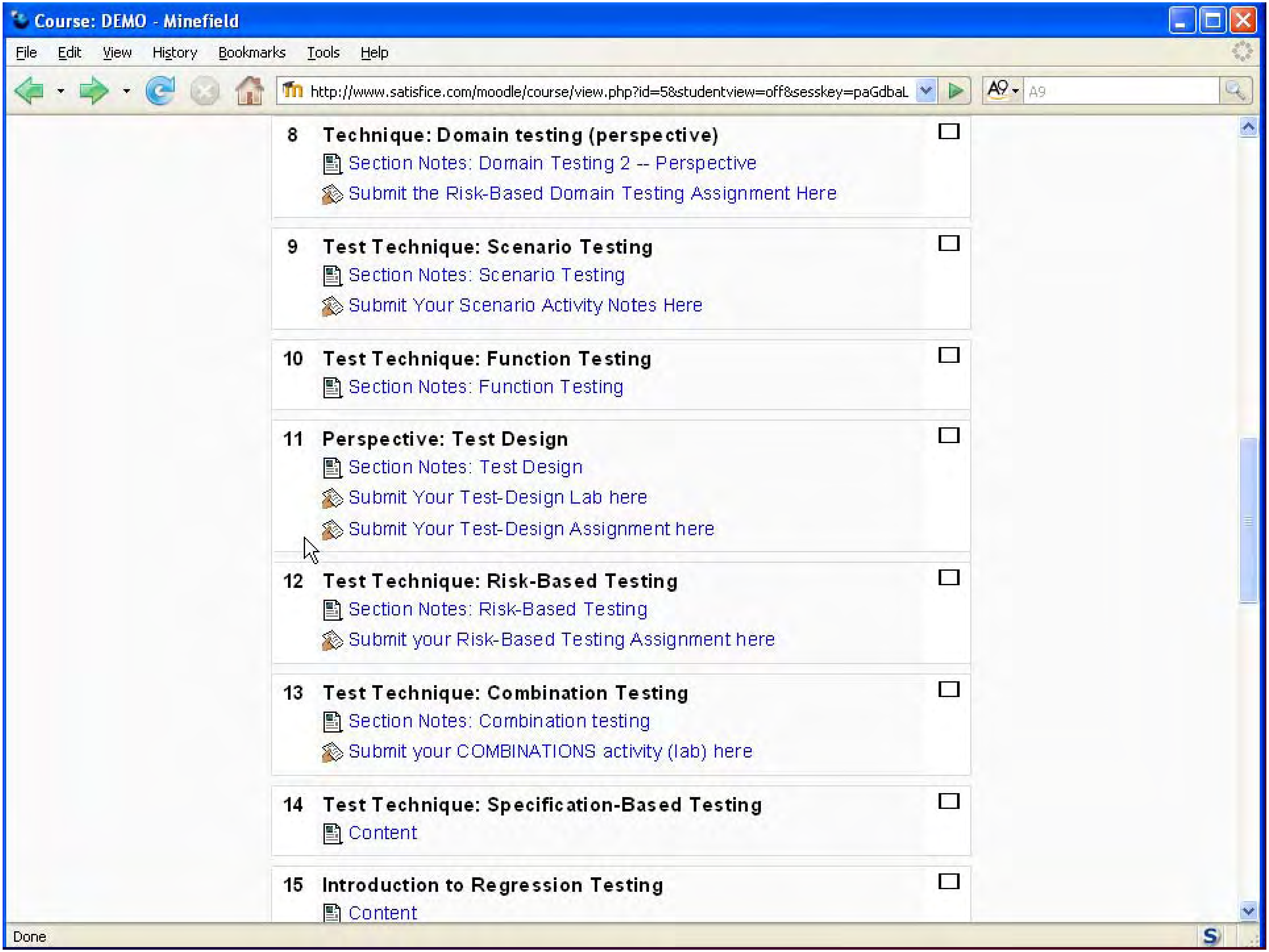
5 **Quality Cost Analysis** □
- Section Notes: Quality Cost Analysis
- Quiz on quality-related costs

6 **Advanced Topics in Bug Advocacy** □
- Section notes: Advanced Topics in Bug Advocacy

7 **Test Technique: Domain Testing** □
- Section Notes: Domain Testing 1 -- Introducing the approach
- Submit the domain testing _lab_ (in-class activity) here
- Submit the domain testing _assignment_ here

8 **Technique: Domain testing (perspective)** □
- Section Notes: Domain Testing 2 -- Perspective

http://www.satisfice.com/moodle/mod/resource/view.php?id=402



Course: DEMO - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/course/view.php?id=5&studentview=off&sesskey=paGdbaL
A9

8 Technique: Domain testing (perspective) □
Section Notes: Domain Testing 2 -- Perspective
Submit the Risk-Based Domain Testing Assignment Here

9 Test Technique: Scenario Testing □
Section Notes: Scenario Testing
Submit Your Scenario Activity Notes Here

10 Test Technique: Function Testing □
Section Notes: Function Testing

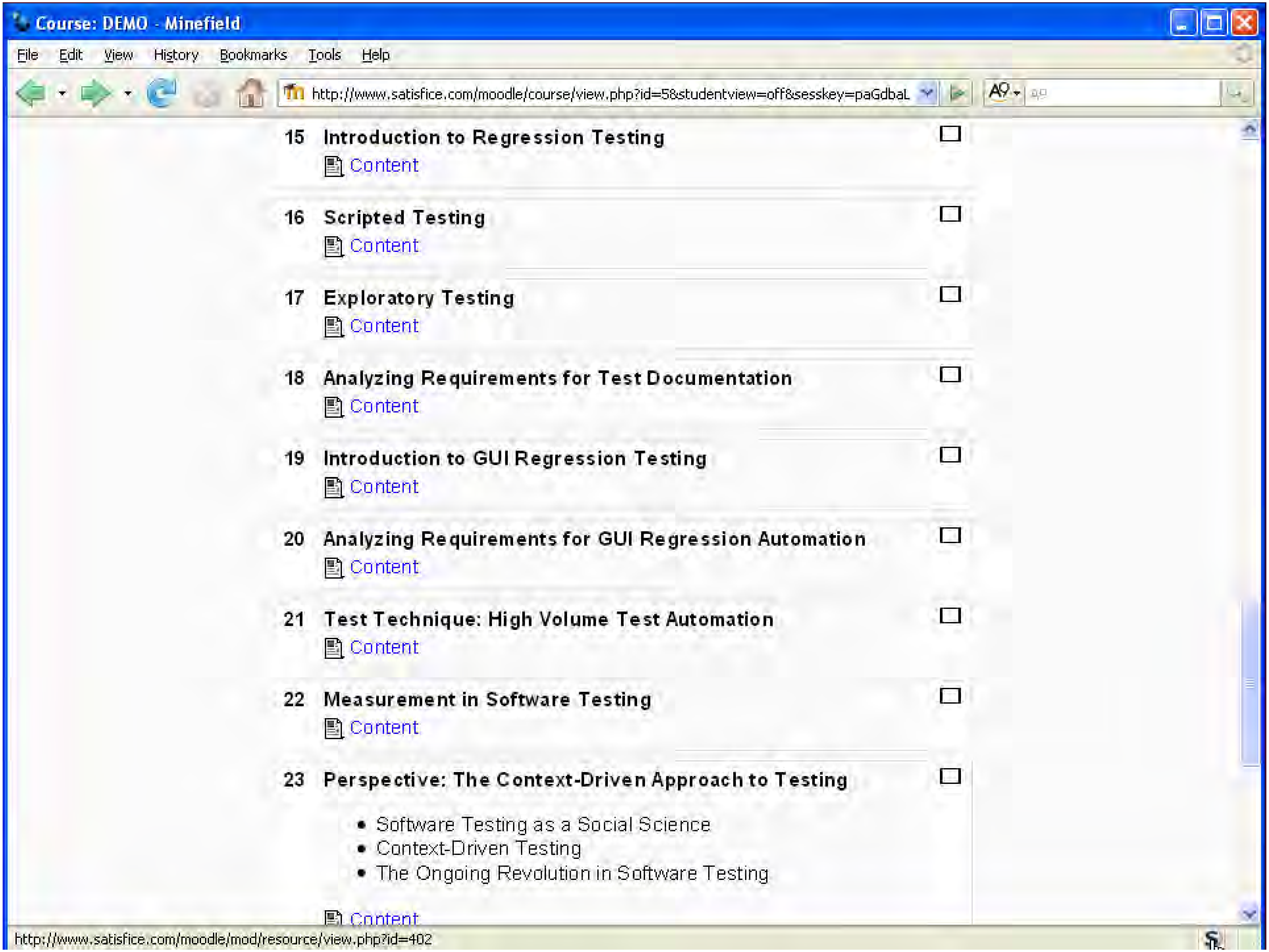
11 Perspective: Test Design □
Section Notes: Test Design
Submit Your Test-Design Lab here
Submit Your Test-Design Assignment here

12 Test Technique: Risk-Based Testing □
Section Notes: Risk-Based Testing
Submit your Risk-Based Testing Assignment here

13 Test Technique: Combination Testing □
Section Notes: Combination testing
Submit your COMBINATIONS activity (lab) here

14 Test Technique: Specification-Based Testing □
Content

15 Introduction to Regression Testing □
Content


Course: DEMO - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/course/view.php?id=5&studentview=off&sesskey=paGdbaL

15 Introduction to Regression Testing □
Content

16 Scripted Testing □
Content

17 Exploratory Testing □
Content

18 Analyzing Requirements for Test Documentation □
Content

19 Introduction to GUI Regression Testing □
Content

20 Analyzing Requirements for GUI Regression Automation □
Content

21 Test Technique: High Volume Test Automation □
Content

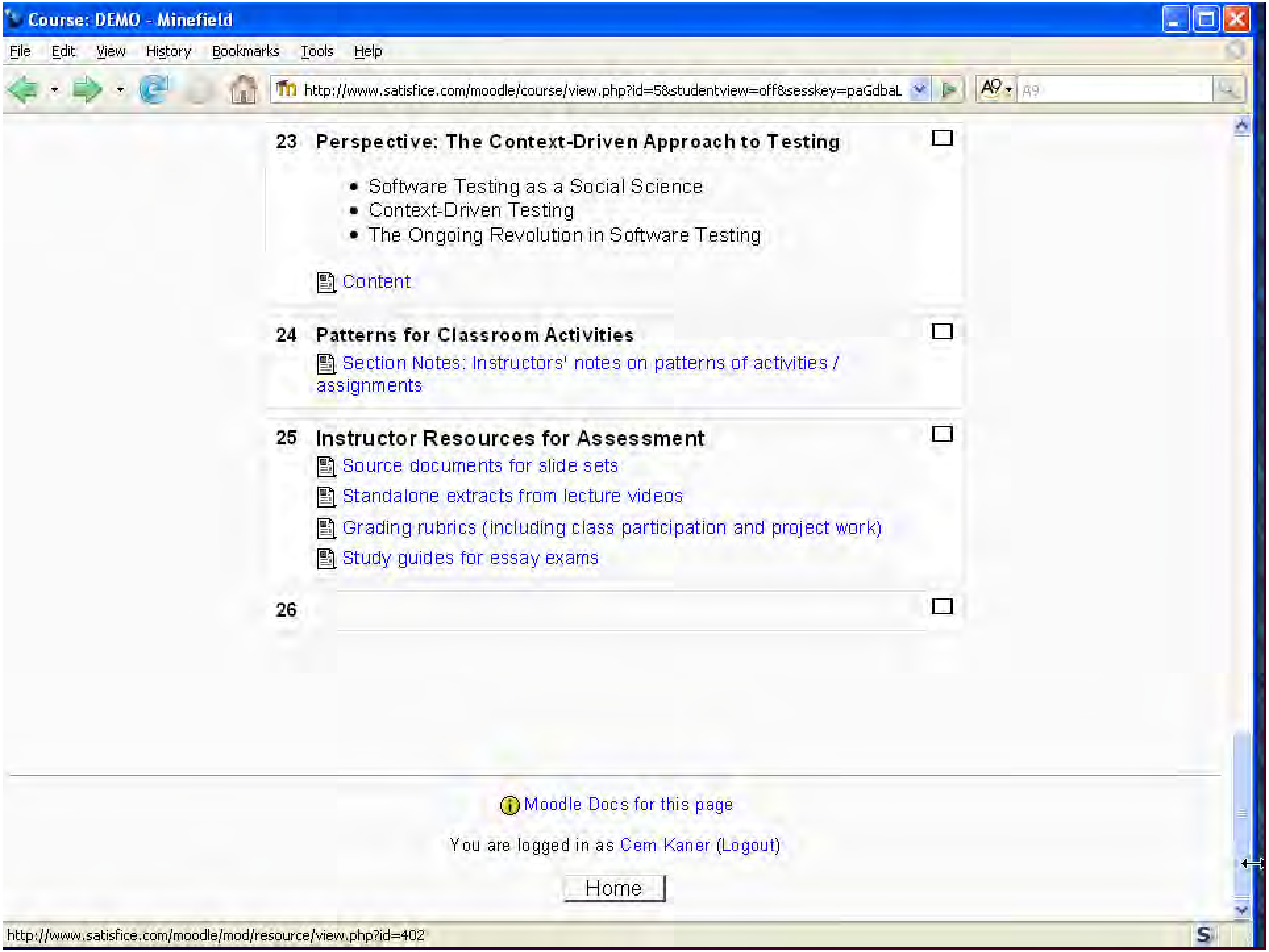
22 Measurement in Software Testing □
Content

23 Perspective: The Context-Driven Approach to Testing □

- Software Testing as a Social Science
- Context-Driven Testing
- The Ongoing Revolution in Software Testing

Content

http://www.satisfice.com/moodle/mod/resource/view.php?id=402


Course: DEMO - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/course/view.php?id=5&studentview=off&sesskey=paGdbaL

23 **Perspective: The Context-Driven Approach to Testing** □

- Software Testing as a Social Science
- Context-Driven Testing
- The Ongoing Revolution in Software Testing

Content

24 **Patterns for Classroom Activities** □

Section Notes: Instructors' notes on patterns of activities / assignments

25 **Instructor Resources for Assessment** □

Source documents for slide sets

Standalone extracts from lecture videos

Grading rubrics (including class participation and project work)

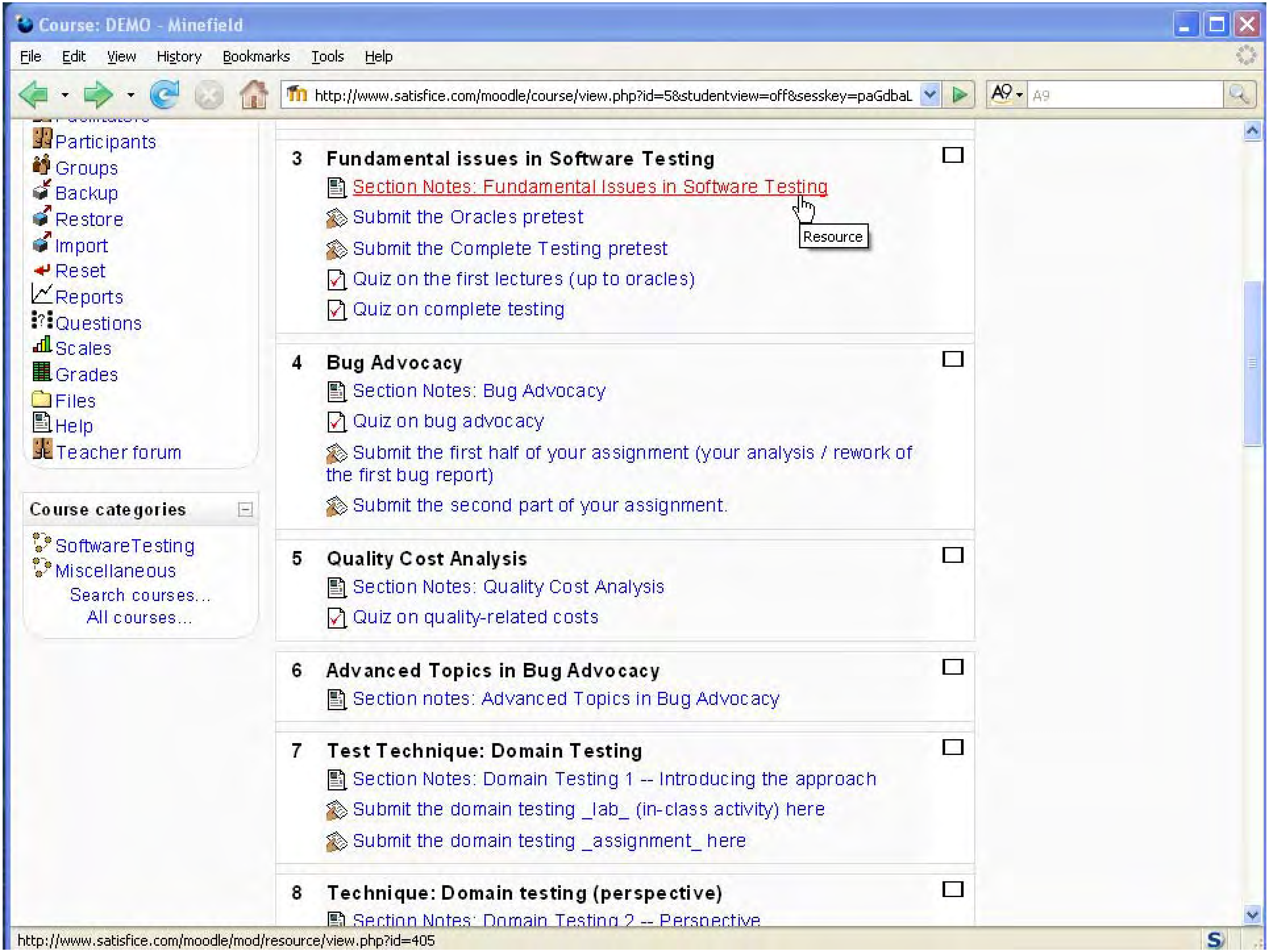
Study guides for essay exams

26 □

Moodle Docs for this page

You are logged in as Cem Kaner (Logout)

Home
http://www.satisfice.com/moodle/mod/resource/view.php?id=402



http://www.satisfice.com/moodle/course/view.php?id=5&studentview=off&sesskey=paGdbaL

3 **Fundamental issues in Software Testing** □

- Section Notes: Fundamental Issues in Software Testing
- Submit the Oracles pretest
- Submit the Complete Testing pretest
- Quiz on the first lectures (up to oracles)
- Quiz on complete testing

4 **Bug Advocacy** □

- Section Notes: Bug Advocacy
- Quiz on bug advocacy
- Submit the first half of your assignment (your analysis / rework of the first bug report)
- Submit the second part of your assignment.

5 **Quality Cost Analysis** □

- Section Notes: Quality Cost Analysis
- Quiz on quality-related costs

6 **Advanced Topics in Bug Advocacy** □

- Section notes: Advanced Topics in Bug Advocacy

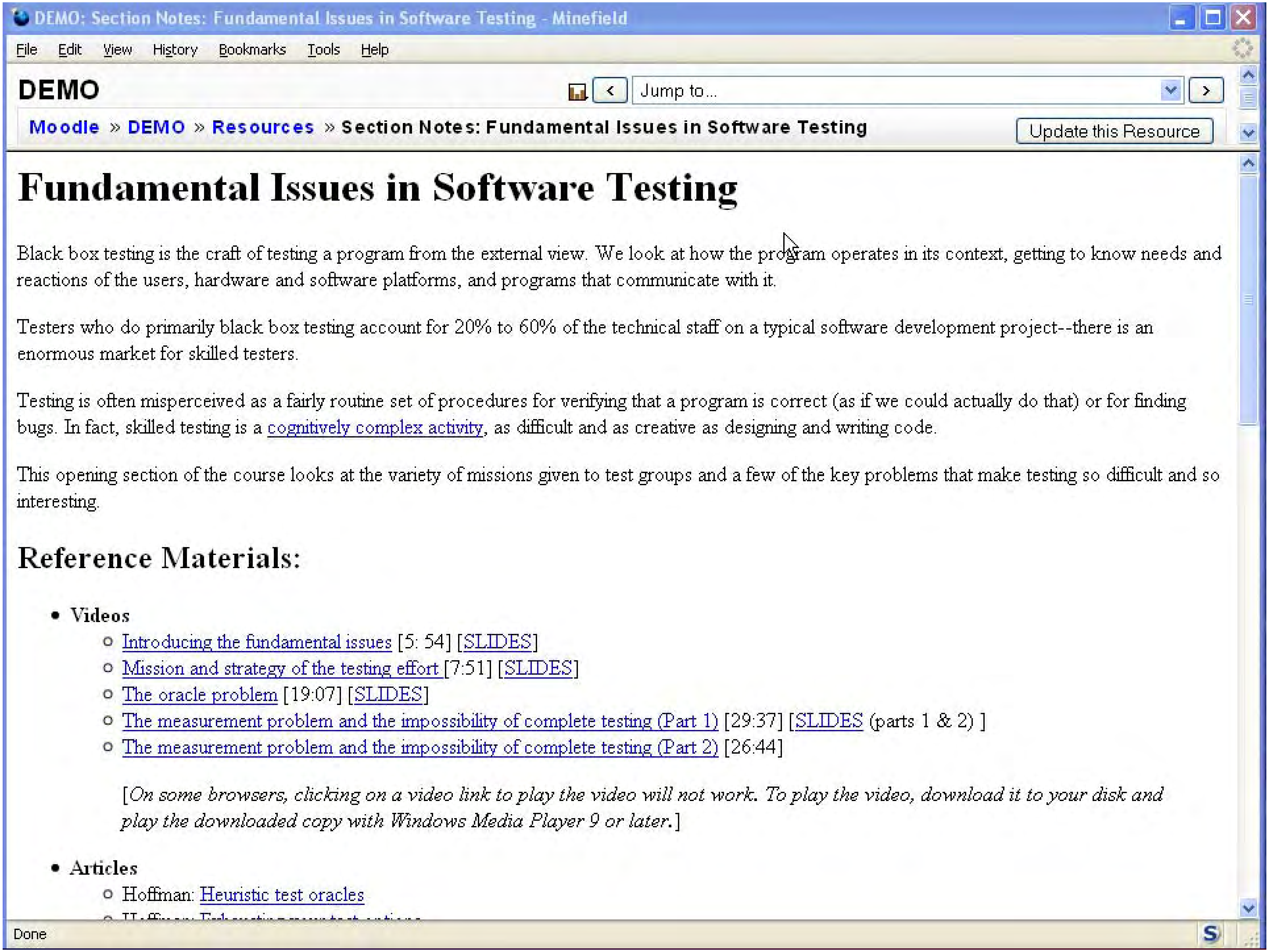
7 **Test Technique: Domain Testing** □

- Section Notes: Domain Testing 1 -- Introducing the approach
- Submit the domain testing _lab_ (in-class activity) here
- Submit the domain testing _assignment_ here

8 **Technique: Domain testing (perspective)** □

- Section Notes: Domain Testing 2 -- Perspective

http://www.satisfice.com/moodle/mod/resource/view.php?id=405


DEMO: Section Notes: Fundamental Issues in Software Testing - Minefield
File
Edit
View
History
Bookmarks
Tools
Help
DEMO
Jump to...
Moodle » DEMO » Resources » Section Notes: Fundamental Issues in Software Testing
Update this Resource
Done

Fundamental Issues in Software Testing

Black box testing is the craft of testing a program from the external view. We look at how the program operates in its context, getting to know needs and reactions of the users, hardware and software platforms, and programs that communicate with it.

Testers who do primarily black box testing account for 20% to 60% of the technical staff on a typical software development project--there is an enormous market for skilled testers.

Testing is often misperceived as a fairly routine set of procedures for verifying that a program is correct (as if we could actually do that) or for finding bugs. In fact, skilled testing is a cognitively complex activity, as difficult and as creative as designing and writing code.

This opening section of the course looks at the variety of missions given to test groups and a few of the key problems that make testing so difficult and so interesting.

Reference Materials:

- **Videos**
  - Introducing the fundamental issues [5: 54] [SLIDES]
  - Mission and strategy of the testing effort [7:51] [SLIDES]
  - The oracle problem [19:07] [SLIDES]
  - The measurement problem and the impossibility of complete testing (Part 1) [29:37] [SLIDES (parts 1 & 2)]
  - The measurement problem and the impossibility of complete testing (Part 2) [26:44]

  [*On some browsers, clicking on a video link to play the video will not work. To play the video, download it to your disk and play the downloaded copy with Windows Media Player 9 or later.*]

- **Articles**
  - Hoffman: Heuristic test oracles
  - Hoffman: Exhausting your test options

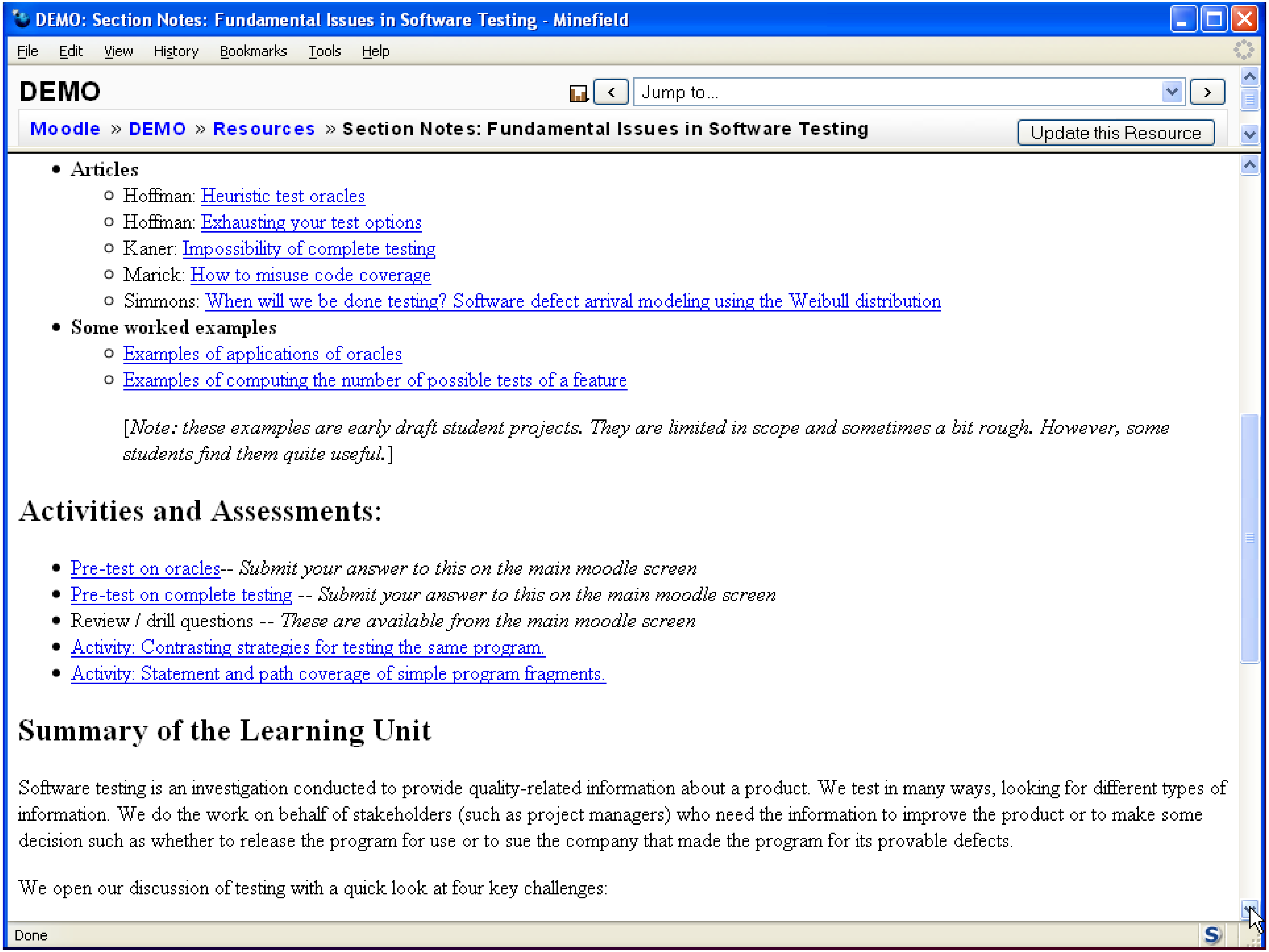
How the Course Works

- *Students watch the video before coming to class*
- *Students often work through an open-book quiz before coming to class*
- *We spend classroom time on*
  - *coached activities*
  - *facilitated discussions*
  - *group feedback (lecture) when I see a class-wide problem*
- *We apply the material in*
  - *in-class activities*
  - *out-of-class assignments*

Lectures On-Line

- *The results seem good*
  - *Good student satisfaction*
  - *Exam results aren’t as different as I expected*
- *Not enough time for the activities*

In an in-house course, time is not constrained by the same type of schedule. It is constrained by value to the project and the staff.


DEMO: Section Notes: Fundamental Issues in Software Testing - Minefield
File Edit View History Bookmarks Tools Help
DEMO
Jump to...
Moodle » DEMO » Resources » Section Notes: Fundamental Issues in Software Testing
Update this Resource
Done

- **Articles**
  - Hoffman: Heuristic test oracles
  - Hoffman: Exhausting your test options
  - Kaner: Impossibility of complete testing
  - Marick: How to misuse code coverage
  - Simmons: When will we be done testing? Software defect arrival modeling using the Weibull distribution
- **Some worked examples**
  - Examples of applications of oracles
  - Examples of computing the number of possible tests of a feature

    [*Note: these examples are early draft student projects. They are limited in scope and sometimes a bit rough. However, some students find them quite useful.*]

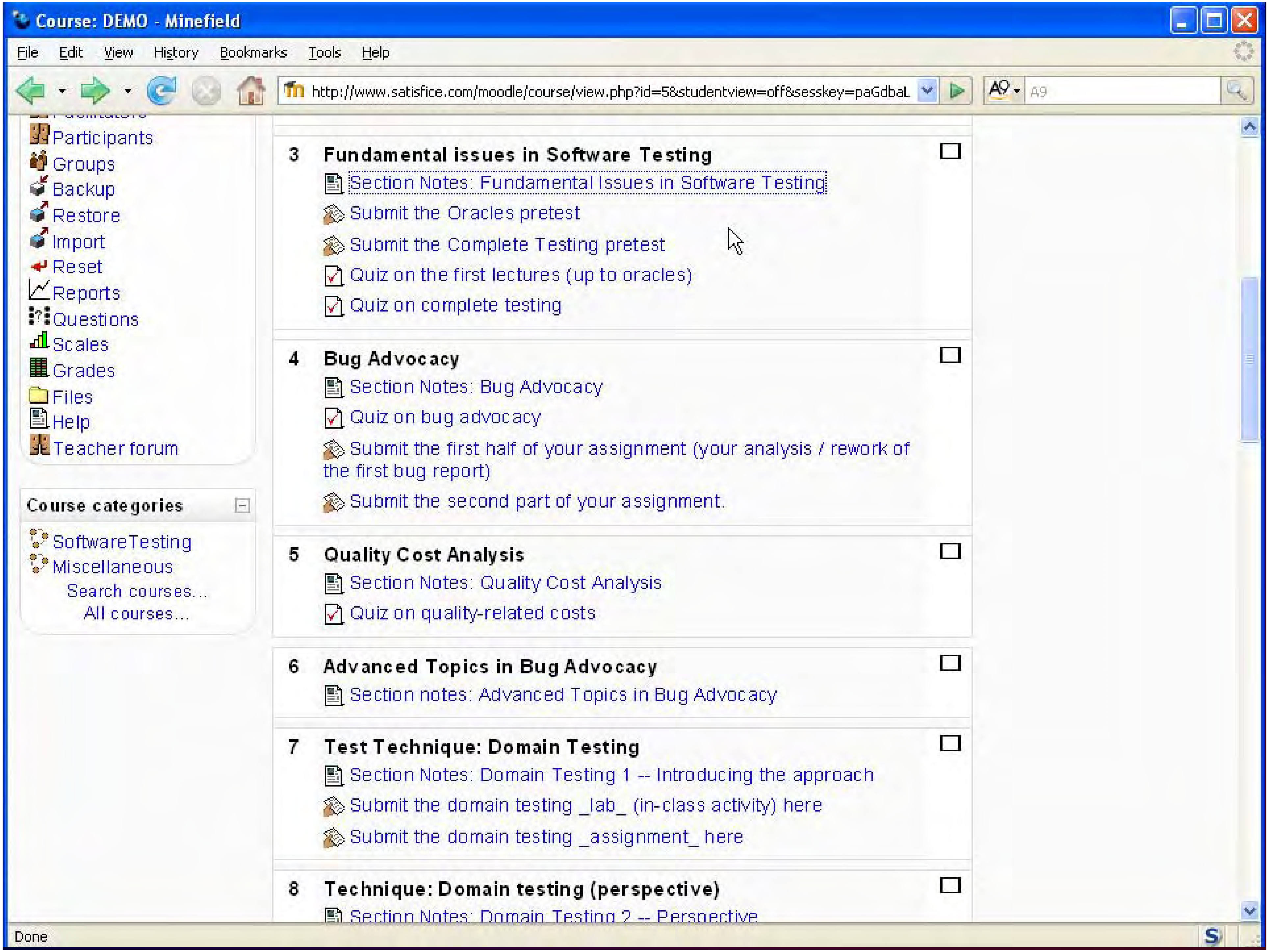
Activities and Assessments:

- Pre-test on oracles-- *Submit your answer to this on the main moodle screen*
- Pre-test on complete testing -- *Submit your answer to this on the main moodle screen*
- Review / drill questions -- *These are available from the main moodle screen*
- Activity: Contrasting strategies for testing the same program.
- Activity: Statement and path coverage of simple program fragments.

Summary of the Learning Unit

Software testing is an investigation conducted to provide quality-related information about a product. We test in many ways, looking for different types of information. We do the work on behalf of stakeholders (such as project managers) who need the information to improve the product or to make some decision such as whether to release the program for use or to sue the company that made the program for its provable defects.

We open our discussion of testing with a quick look at four key challenges:



Course: DEMO - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/course/view.php?id=5&studentview=off&sesskey=paGdbaL
A9
Done

Participants
Groups
Backup
Restore
Import
Reset
Reports
Questions
Scales
Grades
Files
Help
Teacher forum

Course categories

SoftwareTesting
Miscellaneous
Search courses...
All courses...

3 Fundamental issues in Software Testing □

- Section Notes: Fundamental Issues in Software Testing
- Submit the Oracles pretest
- Submit the Complete Testing pretest
- Quiz on the first lectures (up to oracles)
- Quiz on complete testing

4 Bug Advocacy □

- Section Notes: Bug Advocacy
- Quiz on bug advocacy
- Submit the first half of your assignment (your analysis / rework of the first bug report)
- Submit the second part of your assignment.

5 Quality Cost Analysis □

- Section Notes: Quality Cost Analysis
- Quiz on quality-related costs

6 Advanced Topics in Bug Advocacy □

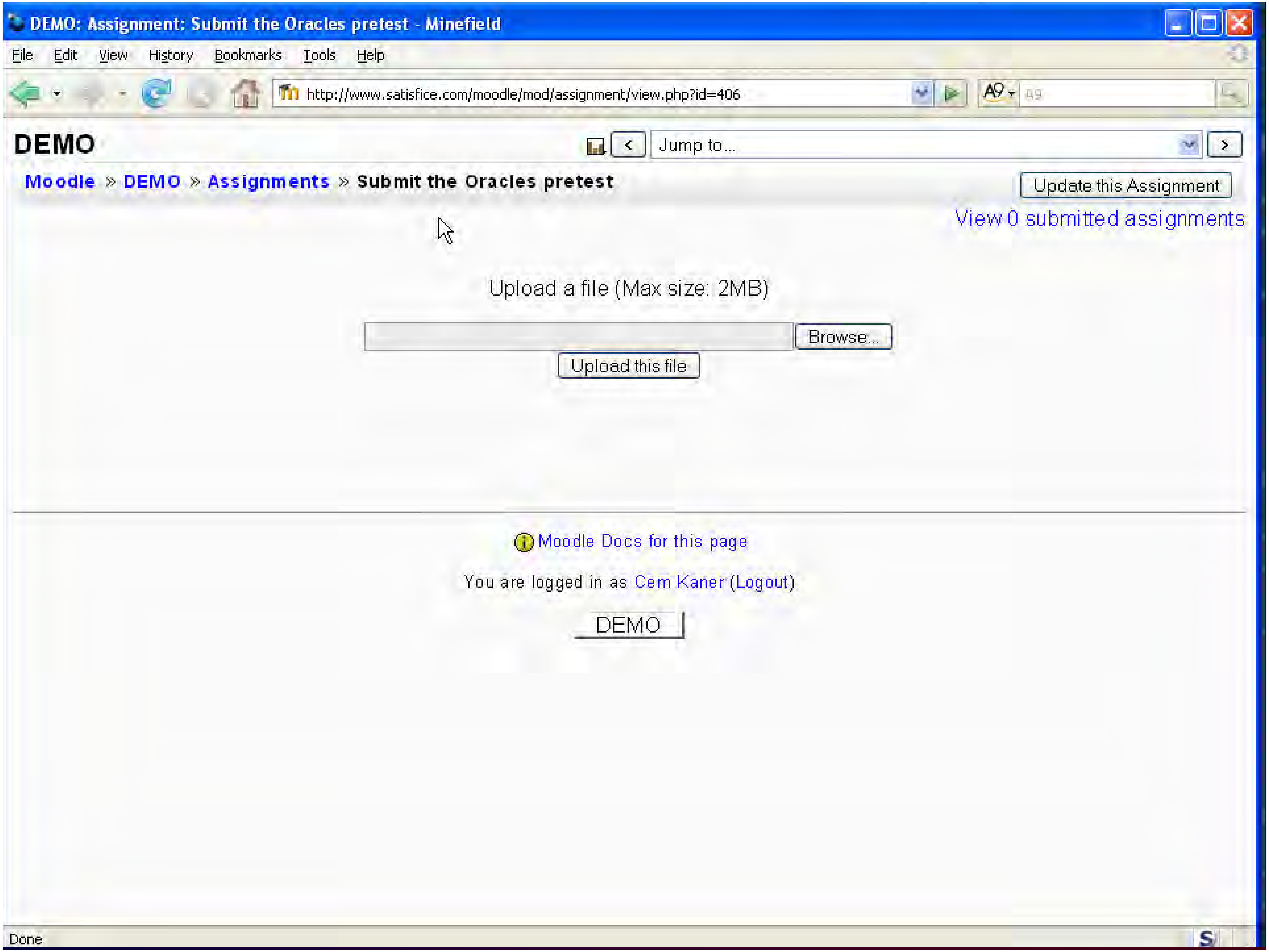
- Section notes: Advanced Topics in Bug Advocacy

7 Test Technique: Domain Testing □

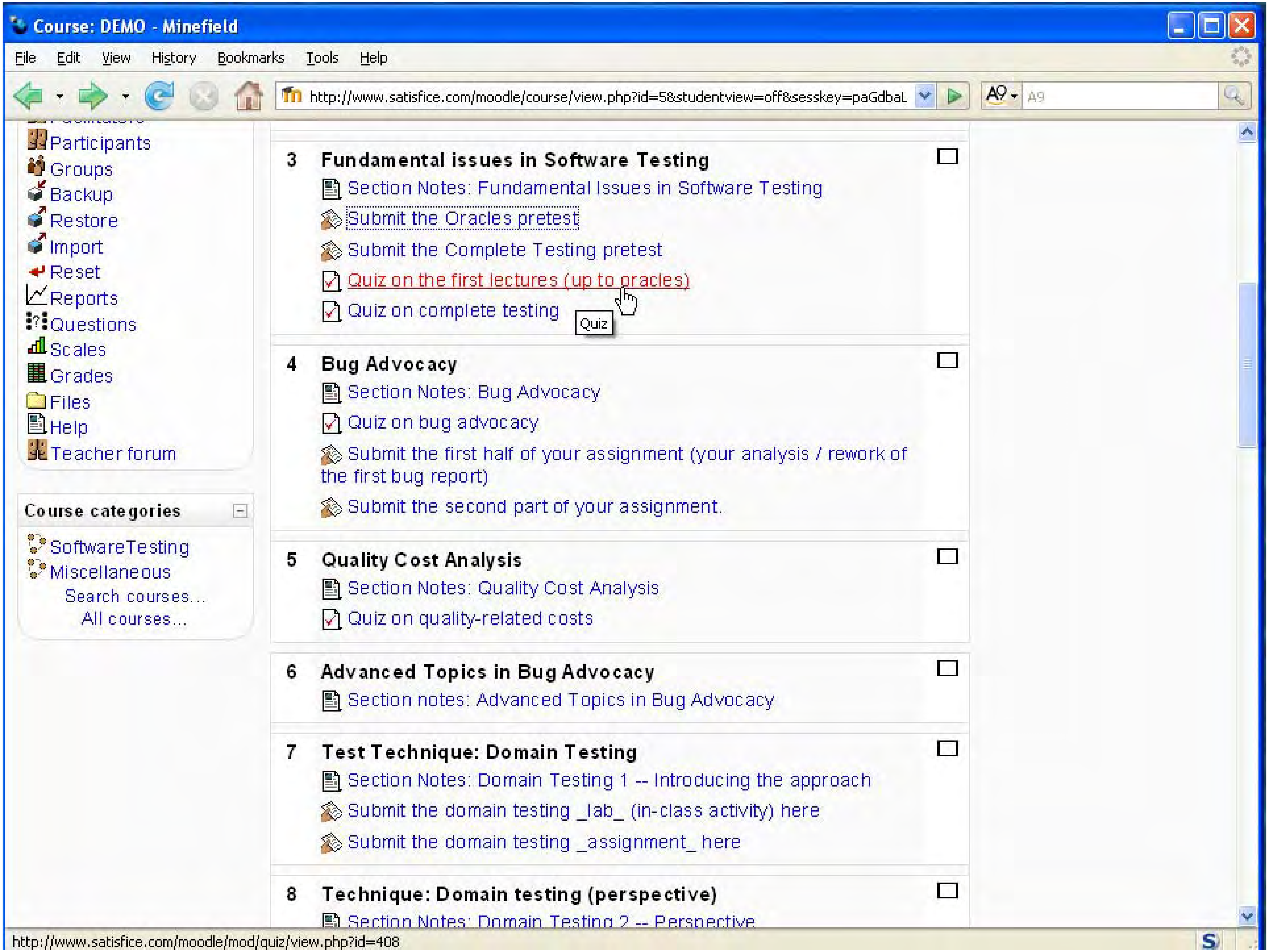
- Section Notes: Domain Testing 1 -- Introducing the approach
- Submit the domain testing _lab_ (in-class activity) here
- Submit the domain testing _assignment_ here

8 Technique: Domain testing (perspective) □

- Section Notes: Domain Testing 2 -- Perspective



DEMO: Assignment: Submit the Oracles pretest - Minefield
File Edit View History Bookmarks Tools Help
http://www.satisfice.com/moodle/mod/assignment/view.php?id=406
DEMO
Jump to...
Moodle » DEMO » Assignments » Submit the Oracles pretest
Update this Assignment
View 0 submitted assignments
Upload a file (Max size: 2MB)
Browse...
Upload this file
Moodle Docs for this page
You are logged in as Cem Kaner (Logout)
DEMO
Done



3 **Fundamental issues in Software Testing** □

- Section Notes: Fundamental Issues in Software Testing
- Submit the Oracles pretest
- Submit the Complete Testing pretest
- Quiz on the first lectures (up to oracles)
- Quiz on complete testing

4 **Bug Advocacy** □

- Section Notes: Bug Advocacy
- Quiz on bug advocacy
- Submit the first half of your assignment (your analysis / rework of the first bug report)
- Submit the second part of your assignment.

5 **Quality Cost Analysis** □

- Section Notes: Quality Cost Analysis
- Quiz on quality-related costs

6 **Advanced Topics in Bug Advocacy** □

- Section notes: Advanced Topics in Bug Advocacy

7 **Test Technique: Domain Testing** □

- Section Notes: Domain Testing 1 -- Introducing the approach
- Submit the domain testing _lab_ (in-class activity) here
- Submit the domain testing _assignment_ here

8 **Technique: Domain testing (perspective)** □

- Section Notes: Domain Testing 2 -- Perspective

Sample Activity: Contrasting Missions

- *Your group is testing a spreadsheet / database. Please consider what your testing strategy should be and what types of test documentation to deliver.*
- *Different groups consider this question:*
  - *Traditional end-of-cycle test group*
  - *Development support near start of project*
  - *Testing a character database for a game*
  - *Testing a custom application for a medical device maker*
- *Groups report back, either by report/discussion to full group or by rotation of group representatives into discussion groups*

Application Under Test

- *We pick a well-known product*
- *Students apply what they learn to that product*
- *Typically, I use an open source product because it avoids NDA problems, students can show their work at interviews*
- *Facilitates student learning (application level and above)*
- *Facilitates student transfer of skills / knowledge to the workplace*

In an in-house course, the AUT is your product

Study Guides

- *www.testingeducation.org/k04/BBSTreviewfall2005.htm*
- *100 questions, include all candidates for mid-term and final exam*
- *Students prepare answers together, assess each other's work*
- *I can require well-organized, thoughtful answers*
- *Fosters strategic preparation*
- *Reduces disadvantage of students whose native language is not English*
- *Creates cooperative learning tasks that should help limited-English-proficiency students improve language skills*

Study Guides

- *Study guide results*
  - *Students inexperienced with these, often blow the first test*
  - *Make-up mid-terms*
    - *Replace grade, not average, not best 1 of 2 results*
    - *Students who take it improve more (1st test compared to final exam) than students who did not take it*
      - *Practice effect, motivation confound*
  - *Writing is better, answers are better, I have greater freedom to grade less forgivingly*
  - *Many students told me this was the most valuable learning experience in the course, and the most time-consuming*

Study Guides

- *In-house use:*
  - *Focus discussion of course materials*
  - *Potential interview questions, especially if you revise them to apply to your class of product*

Assessing student reaction

- Chose the Student Assessment of Learning Gains http://www.flaguide.org/cat/salg/
- Measures student perceptions of their 'gains' in learning
- Customizable
- Administered online
- FREE
- Beats the standard course evaluation form!
- Students each spent over an hour providing their evaluation.

Overview

1. *Tour of the Moodle course management system*
2. *Tour of the Black Box Software Testing Course on Moodle*
3. *Overview of use of material like this in the workplace*
4. *Dealing with the instructional challenges of a cognitively complex field of study*
5. *Taking control of your learning objectives for the course*
6. *Examples of activity patterns*

Instruction in the Workplace

- *The opportunity:*
  - *Build on the strengths of commercial instruction*
  - *Avoid the weaknesses of academic instruction*

Commercial vs. Academic

• *Drive-by teaching* – *2-5 days, rapid-fire ideas, visiting instructor*	• *Local teaching* – *Several months, a few hours per week, students get to know instructor*
• *Broad, shallow coverage*	• *Deeper coverage*
• *Time constraints limit activities*	• *Activities expected to develop skills*
• *No time for homework*	• *Extensive homework*
• *No exams*	• *Assessment expected*
• *Coached, repeated practice seen as time-wasting*	• *Coached, repeated practice is highly appreciated*
• *Familiarity*	• *Capability*
• *Work experience helps to bring home concepts*	• *Students have no work experience, need context*
• *Richer grounding in real practice*	• *Harder to connect to real practice*
• *Some (occasional) student groups share a genuine current need*	• *Students don't naturally come to a course as a group with a shared problem*
• *Objective: one applicable new idea per day*	• *Expect mastery of several concepts and skills*

Build on the Strengths

- *Adopt a deliberate, slow pace*
- *Work as a learning team*
- *Focus on application of the tasks to current projects, use the course as a vehicle for tinkering with productivity and creativity of your day-to-day work*

Demonstrating current value of the learning experience builds management support for continued investment

One vision of the in-house course

- *Meet weekly for a year*
- *Watch 10-25 minutes of video in advance*
- *Discuss the lesson and its applicability*
- *Over the next week, try to apply it on the job to the current project(s) in test*
- *Discuss the application results in the next week or move to the next segment.*
- *At the end of the course, students know how things fit into their environment, and have multiple examples (from multiple student colleagues).*

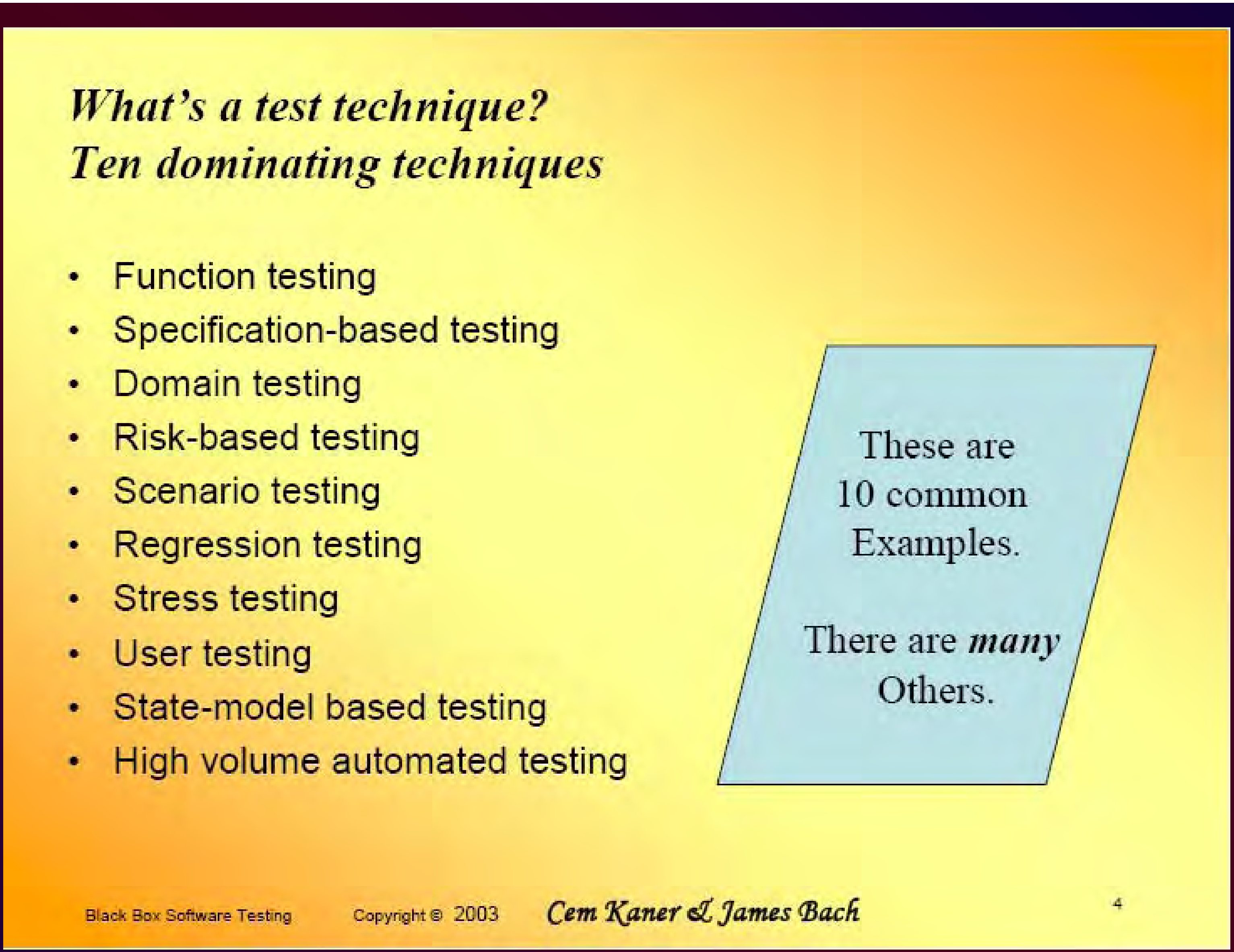
Overview

1. *Tour of the Moodle course management system*
2. *Tour of the Black Box Software Testing Course on Moodle*
3. *Overview of use of material like this in the workplace*
4. ***Dealing with the instructional challenges of a cognitively complex field of study***
5. *Taking control of your learning objectives for the course*
6. *Examples of activity patterns*

The instructional challenge, as I see it

Software testing
is cognitively complex,
requires critical thinking,
effective communication, and
rapid self-directed learning.

Software testing is a process of empirical, technical investigation of the product under test conducted to provide stakeholders with quality-related information.



What's a test technique?
Ten dominating techniques

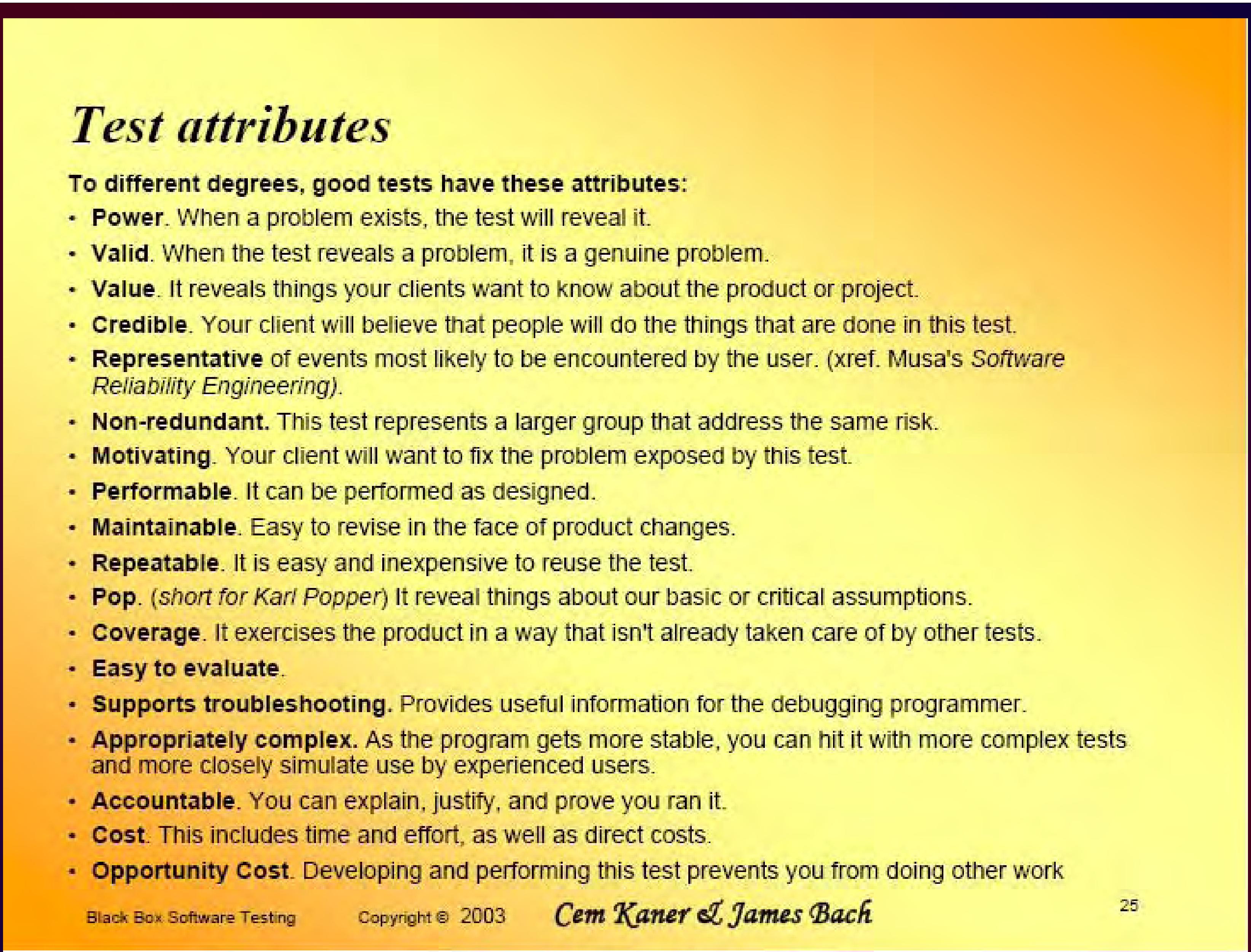
- Function testing
- Specification-based testing
- Domain testing
- Risk-based testing
- Scenario testing
- Regression testing
- Stress testing
- User testing
- State-model based testing
- High volume automated testing

These are
10 common
Examples.

There are ***many***
Others.

Black Box Software Testing Copyright © 2003 *Cem Kaner & James Bach* 4

http://www.testingeducation.org/BBST/BBST--IntroductiontoTestDesign.html



Test attributes

To different degrees, good tests have these attributes:

- **Power**. When a problem exists, the test will reveal it.
- **Valid**. When the test reveals a problem, it is a genuine problem.
- **Value**. It reveals things your clients want to know about the product or project.
- **Credible**. Your client will believe that people will do the things that are done in this test.
- **Representative** of events most likely to be encountered by the user. (xref. Musa's *Software Reliability Engineering*).
- **Non-redundant.** This test represents a larger group that address the same risk.
- **Motivating**. Your client will want to fix the problem exposed by this test.
- **Performable**. It can be performed as designed.
- **Maintainable**. Easy to revise in the face of product changes.
- **Repeatable**. It is easy and inexpensive to reuse the test.
- **Pop**. (*short for Karl Popper*) It reveal things about our basic or critical assumptions.
- **Coverage**. It exercises the product in a way that isn't already taken care of by other tests.
- **Easy to evaluate**.
- **Supports troubleshooting.** Provides useful information for the debugging programmer.
- **Appropriately complex.** As the program gets more stable, you can hit it with more complex tests and more closely simulate use by experienced users.
- **Accountable**. You can explain, justify, and prove you ran it.
- **Cost**. This includes time and effort, as well as direct costs.
- **Opportunity Cost**. Developing and performing this test prevents you from doing other work

http://www.testingeducation.org/BBST/BBST--IntroductiontoTestDesign.html

Contexts Vary Across Projects

Testers must learn, for each new product:

- *What are the goals and quality criteria for the project*
- *What skills and resources are available to the project*
- *What is in the product*
- *How it could fail*
- *What the consequences of potential failures could be*
- *Who might care about which consequence of what failure*
- *How to trigger a fault that generates the failure we're seeking*
- *How to recognize failure*
- *How to decide what result variables to pay attention to*
- *How to decide what* other *result variables to pay attention to in the event of intermittent failure*
- *How to troubleshoot and simplify a failure, so as to better*
  - *(a) motivate a stakeholder who might advocate for a fix*
  - *(b) enable a fixer to identify and stomp the bug more quickly*
- *How to expose, and who to expose to, undelivered benefits, unsatisfied implications, traps, and missed opportunities.*

It's kind of like CSI

AFIS
Agar
Alarm pen
Alginate
ALS (Alternate light source)
Amido Black
Ammeter
Analytical balance
Analytical scale
Anechoic chamber
Angiogram
Autoclave
Ballistic gelatin
Biohazard bag
Bloody print enhancer
Blower brush
Boxed reference ammunition collection
Bromoform

AFIS

Automated Fingerprint Identification System: Computer network that scans crime-scene fingerprints and compares them with millions of prints collected by law enforcement agencies around the state, region, country, and world. A print is traced by a fingerprint expert. The tracing is then scanned by the computer, which assigns values to various features of the print. Those values are compared to other

CSI HANDBOOK

Learn more about tools, evidence and procedures used by CSIs.

EPISODES

Missed an episode? Catch up on previous stories now.

PREVIOUSLY ON CSI:

Room Service

Julian Harper (23), touted as the next Brad Pitt, is taking his bright lights moment to the max. Sex, drugs and a "High Roller Suite" are put at his disposal as he enters the ...more

CSI NEWSLETTER

Be among the first to know. Subscribe now for email updates

CSI: VIDEO

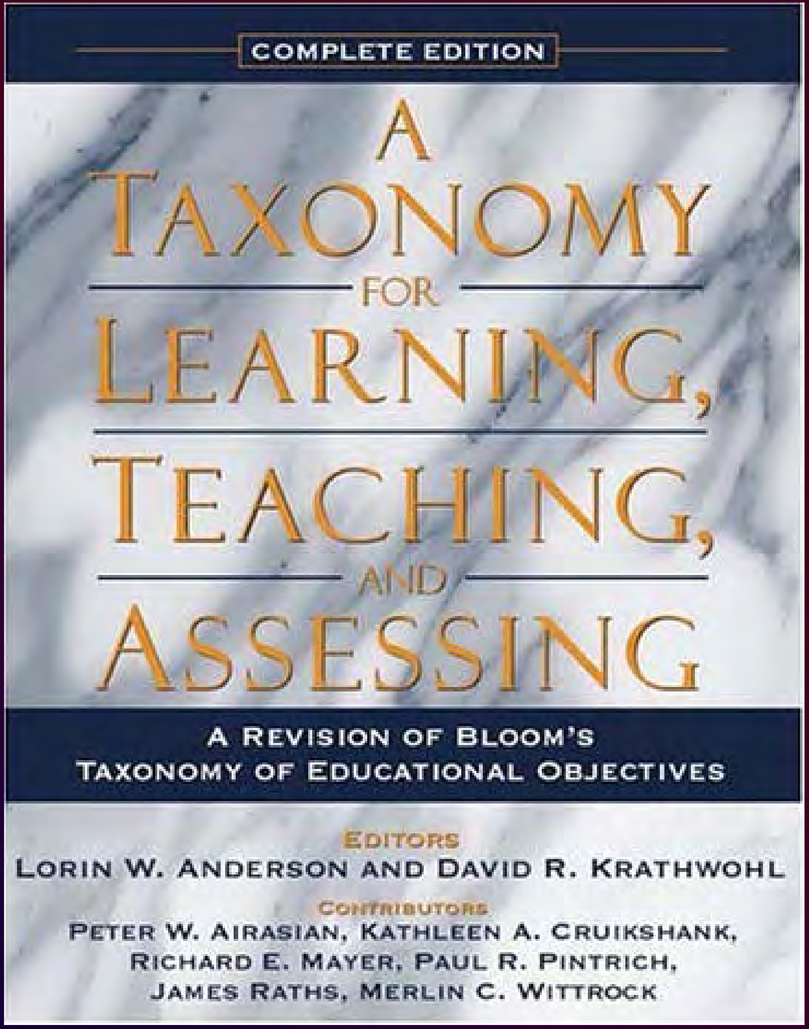
Behind the Scenes

"CSI" celebrates 100 episodes

MANY tools, procedures, sources of evidence.

- Tools and procedures don't define an investigation or its goals.
- There is too much evidence to test, tools are often expensive, so investigators must exercise judgment.
- The investigator must pick what to study, and how, in order to reveal the most needed information.

Characterizing Cognitive Complexity



COMPLETE EDITION

A TAXONOMY FOR LEARNING, TEACHING, AND ASSESSING

A REVISION OF BLOOM'S TAXONOMY OF EDUCATIONAL OBJECTIVES

EDITORS

LORIN W. ANDERSON AND DAVID R. KRATHWOHL

CONTRIBUTORS

PETER W. AIRASIAN, KATHLEEN A. CRUIKSHANK,
RICHARD E. MAYER, PAUL R. PINTRICH,
JAMES RATHS, MERLIN C. WITTROCK

- *Anderson & Krathwohl (2001) provide a modern update to Bloom's (1956) taxonomy*

Characterizing Cognitive Complexity

		Cognitive Process Dimension					
		Remember	Understand	Apply	Analyze	Evaluate	Create
Knowledge Dimension	Factual	**lecture**	**lecture**				
	Conceptual	**lecture**	**lecture**				
	Procedural	**lecture**	**lecture**				
	Meta-Cognitive						

Anderson & Krathwohl, 2001

A Slight Variation for Testing

- *Facts*
- *Concepts*
- *Procedures*
- *Cognitive strategies*
- *Models*
- *Skills*
- *Attitudes*
- *Metacognition*

- *The Testing Learning Concepts Taxonomy, Kaner & Bach, unpublished beta version*

Variation for Testing: Facts

- A "statement of fact" is a statement that can be unambiguously proved true or false. For example, "James Bach was born in 1623" is a statement of fact. (But not true, for the James Bach we know and love.) A fact is the subject of a true statement of fact.
- Facts include such things as:
  - Tidbits about famous people
  - Famous examples (the example might also be relevant to a concept, procedure, skill or attitude)
  - Items of knowledge about devices (for example, a description of an interoperability problem between two devices)

Variation for Testing: Concepts

- A concept is a general idea. "Concepts are abstract in that they omit the differences of things in their extension, treating them as if they were identical." (wikipedia: Concept).
- In practical terms, we treat the following kinds of things as "concepts" in this taxonomy:
  - definitions
  - descriptions of relationships between things
  - descriptions of contrasts between things
  - description of the idea underlying a practice, process, task, heuristic (whatever)
- Here's a distinction that you might find useful.
  - Consider the oracle heuristic, "Compare the behavior of this program with a respected competitor and report a bug if this program's behavior seems inconsistent with and possibly worse than the competitor's."
    - If I am merely describing the heuristic, I am giving you a concept.
    - If I tell you to make a decision based on this heuristic, I am giving you a
- Sometimes, a rule is a concept.
  - A rule is an imperative ("Stop at a red light") or a causal relationship ("Two plus two yields four") or a statement of a norm ("Don't wear undershorts outside of your pants at formal meetings").
  - The description / definition of the rule is the concept
  - Applying the rule in a straightforward way is application of a concept
  - The decision to puzzle through the value or applicability of a rule is in the realm of cognitive strategies.
  - The description of a rule in a formalized way is probably a model.

Variation for Testing: Procedures

- "Procedures" are algorithms. They include a reproducible set of steps for achieving a goal.
- Consider the task of reporting a bug. Imagine that someone has
  - broken this task down into subtasks (simplify the steps, look for more general conditions, write a short descriptive summary, etc.)
  - and presented the tasks in a sequential order.
- This description is intended as a procedure if the author expects you to do all of the steps in exactly this order every time.
- This description is a cognitive strategy if it is meant to provide a set of ideas to help you think through what you have to do for a given bug, with the understanding that you may do different things in different orders each time, but find this a useful reference point as you go.

Variation for Testing: Cognitive Strategies

"Cognitive strategies are guiding procedures that students can use to help them complete less-structured tasks such as those in reading comprehension and writing. The concept of cognitive strategies and the research on cognitive strategies represent the third important advance in instruction.

There are some academic tasks that are "well-structured." These tasks can be broken down into a fixed sequence of subtasks and steps that consistently lead to the same goal. The steps are concrete and visible. There is a specific, predictable algor ithm that can be followed, one that enables students to obtain the same result each time they perform the algorithmic operations. These well-structured tasks are taught by teaching each step of the algorithm to students. The results of the research on tea cher effects are particularly relevant in helping us learn how teach students algorithms they can use to complete well-structured tasks.

In contrast, reading comprehension, writing, and study skills are examples of less- structured tasks -- tasks that cannot be broken down into a fixed sequence of subtasks and steps that consistently and unfailingly lead to the goal. Because these ta sks are less-structured and difficult, they have also been called higher-level tasks. These types of tasks do not have the fixed sequence that is part of well-structured tasks. One cannot develop algorithms that students can use to complete these tasks."

Gleefully pilfered from: Barak Rosenshine, Advances in Research on Instruction, Chapter 10 in J.W. Lloyd, E.J. Kameanui, and D. Chard (Eds.) (1997) Issues in educating students with disabilities. Mahwah, N.J.: Lawrence Erlbaum: Pp. 197-221. http://epaa.asu.edu/barak/barak.html

In cognitive strategies, we include:

- heuristics (fallible but useful decision rules)
- guidelines (fallible but common descriptions of how to do things)
- good (rather than "best" practices)

The relationship between cognitive strategies and models:

- deciding to apply a model and figuring out how to apply a model involve cognitive strategies
- deciding to create a model and figuring out how to create models to represent or simplify a problem involve cognitive strategies

BUT

- the model itself is a simplified representation of something, done to give you insight into the thing you are modeling.

We aren't sure that the distinction between models and the use of them is worthwhile, but it seems natural to us so we're making it.

Variation for Testing: Models

A model is

- A simplified representation created to make something easier to understand, manipulate or predict some aspects of the modeled object or system.
- Expression of something we don't understand in terms of something we (think we) understand.

- A state-machine representation of a program is a model.
- Deciding to use a state-machine representation of a program as a vehicle for generating tests is a cognitive strategy.
- Slavishly following someone's step-by-step catalog of best practices for generating a state- machine model of a program in order to derive scripted test cases for some fool to follow is a procedure.
- This definition of a model is a concept.
- The assertion that Harry Robinson publishes papers on software testing and models is a statement of fact.

Sometimes, a rule is a model.

- A rule is an imperative ("Stop at a red light") or a causal relationship ("Two plus two yields four") or a statement of a norm ("Don't wear undershorts outside of your pants at formal meetings").
- A description / definition of the rule is probably a concept
- A symbolic or generalized description of a rule is probably a model.

Variation for Testing: Skills

Skills are things that improve with practice.

- Effective bug report writing is a skill, and includes several other skills.
- Taking a visible failure and varying your test conditions until you find a simpler set of conditions that yields the same failure is skilled work. You get better at this type of thing over time.

- Entries into this section will often be triggered by examples (in instructional materials) that demonstrate skilled work, like "Here's how I use this technique" or "Here's how I found that bug."
- The "here's how" might be classed as a:
  - procedure
  - cognitive strategy, or
  - skill
- In many cases, it would be accurate and useful to class it as both a skill and a cognitive strategy.

Variation for Testing: Attitudes

- "An attitude is a persisting state that modifies an individual's choices of action." Robert M. Gagne, Leslie J. Briggs & Walter W. Wager (1992) "Principles of Instructional Design" (4th Ed),, p. 48.
- Attitudes are often based on beliefs (a belief is a proposition that is held as true whether it has been verified true or not).
- Instructional materials often attempt to influence the student's attitudes.
- For example, when we teach students that complete testing is impossible, we might spin the information in different ways to influence student attitudes toward their work:
  - given the impossibility, testers must be creative and must actively consider what they can do at each moment that will yield the highest informational return for their project
  - given the impossibility, testers must conform to the carefully agreed procedures because these reflect agreements reached among the key stakeholders rather than diverting their time to the infinity of interesting alternatives
- Attitudes are extremely controversial in our field and refusal to acknowledge legitimate differences (or even the existence of differences) has been the source of a great deal of ill will.
- In general, if we identify an attitude or an attitude-related belief as something to include as an assessable item, we should expect to create questions that:
  - define the item without requiring the examinee to agree that it is true or valid
  - contrast it with a widely accepted alternative, without requiring the examinee to agree that it is better or preferable to the alternative
  - adopt it as the One True View, but with discussion notes that reference the controversy about this belief or attitude and make clear that this item will be accepted for some exams and bounced out of others.

Variation for Testing: Metacognition

- Metacognition refers to the executive process that is involved in such tasks as:
  - planning (such as choosing which procedure or cognitive strategy to adopt for a specific task)
  - estimating how long it will take (or at least, deciding to estimate and figuring out what skill / procedure / slave-labor to apply to obtain that information)
  - monitoring how well you are applying the procedure or strategy
  - remembering a definition or realizing that you don't remember it and rooting through Google for an adequate substitute
- Much of context-driven testing involves metacognitive questions:
  - which test technique would be most useful for exposing what information that would be of what interest to who?
  - what areas are most critical to test next, in the face of this information about risks, stakeholder priorities, available skills, available resources?
- Questions / issues that should get you thinking about metacognition are:
  - How to think about …
  - How to learn about …
  - How to talk about …
- In the BBST course, the section on specification analysis includes a long metacognitive digression into active reading and strategies for getting good information value from the specification fragments you encounter, search for, or create.

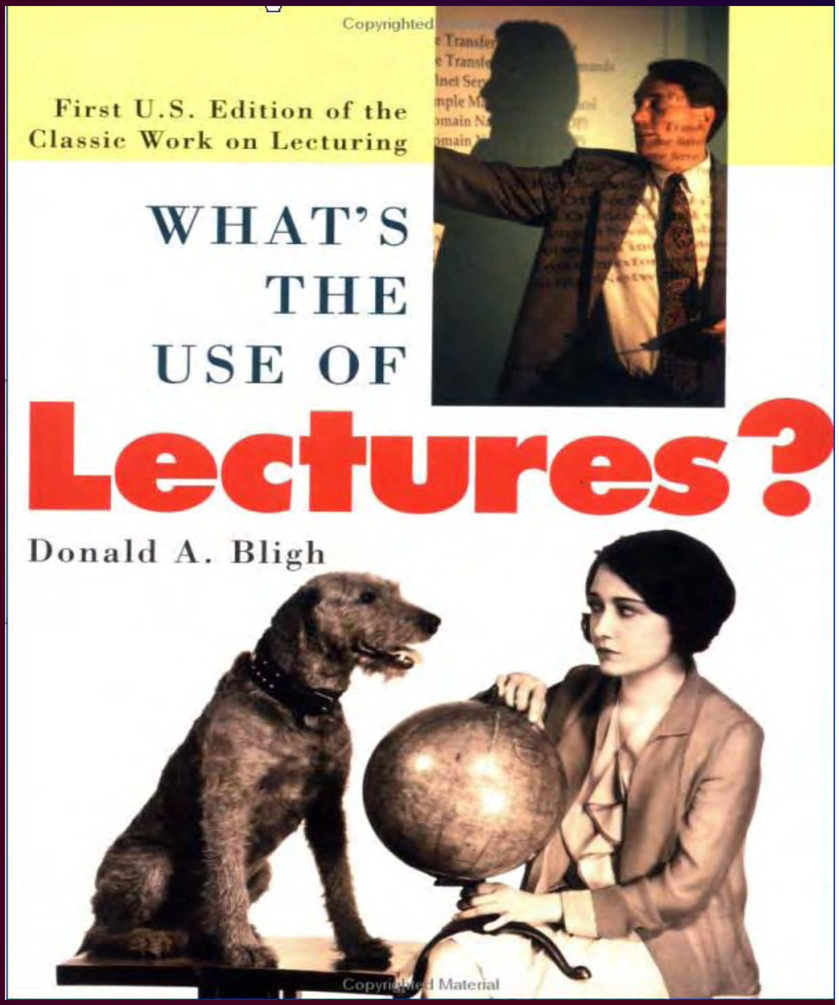
Characterizing Cognitive Complexity

Knowledge Dimension		Cognitive Process Dimension					
		Remember	Understand	Apply	Analyze	Evaluate	Create
	Factual	**lecture**	**lecture**				
	Conceptual	**lecture**	**lecture**				
	Procedural	**lecture**	**lecture**				
	Meta- Cognitive						

Anderson & Krathwohl, 2001

Commercial Teaching Style

- *Primary communication style was lecture*
  - *Real-life examples*
    - *Motivating*
    - *Memorable*
    - *Illustrate applications*
    - *Illustrate complexity*
- *Lectures can be excellent for conveying basic knowledge, but they are weak for developing higher order cognitive skills*


Copyrighted
First U.S. Edition of the Classic Work on Lecturing
WHAT'S THE USE OF Lectures?
Donald A. Bligh
Copyrighted Material

Levels of Learning

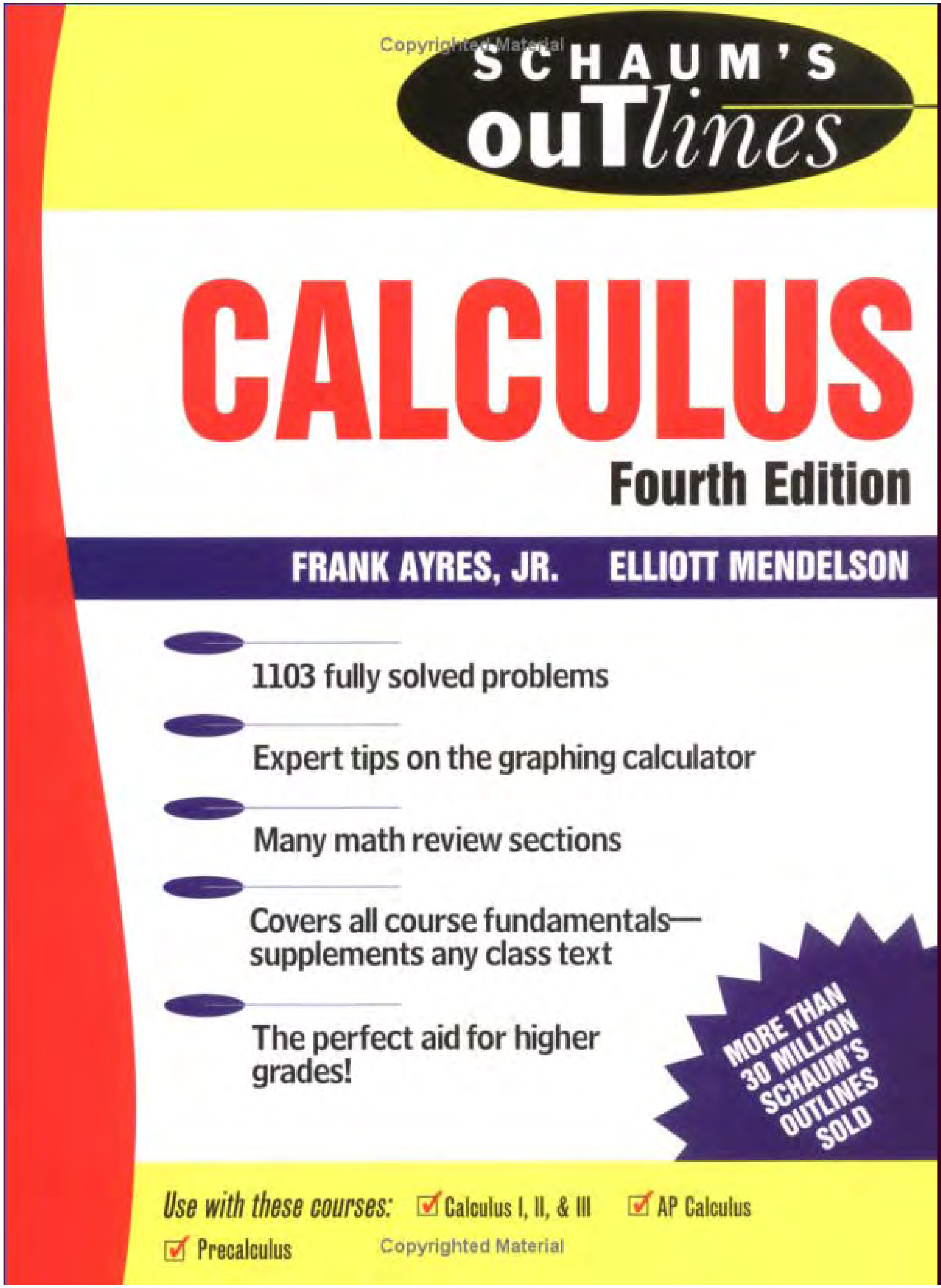
- *Remember*
- *Understand*
  - *It is easy to "teach" at these levels and to assess / evaluate at them.*
  - *Most "objective" tests assess at these levels*
- *Apply*
- *Analyze*
- *Create*
  - *Most professional work is done at these levels.*
  - *We have a transfer problem. Will teaching to the lower levels transfer to the higher?*

Example Problem: Domain Testing

- *Most widely taught testing technique*
  - *For details, see* http://www.testingeducation.org/BBST/Domain.html
  - *Easy to explain the basic concepts*
  - *Classic examples widely taught*
  - *Students quickly signal that they understand it*
  - *But when you give them exercises under slightly new circumstances*
    - *They blow it*
      - *And then they blow the next one*
        - *And the next one . . .*

Brilliant (?) idea

- *Lots of practice exercises*
- *Like we used to do as math students*



Copyrighted Material

SCHAUM'S ouTlines

CALCULUS

Fourth Edition

FRANK AYRES, JR. ELLIOTT MENDELSON

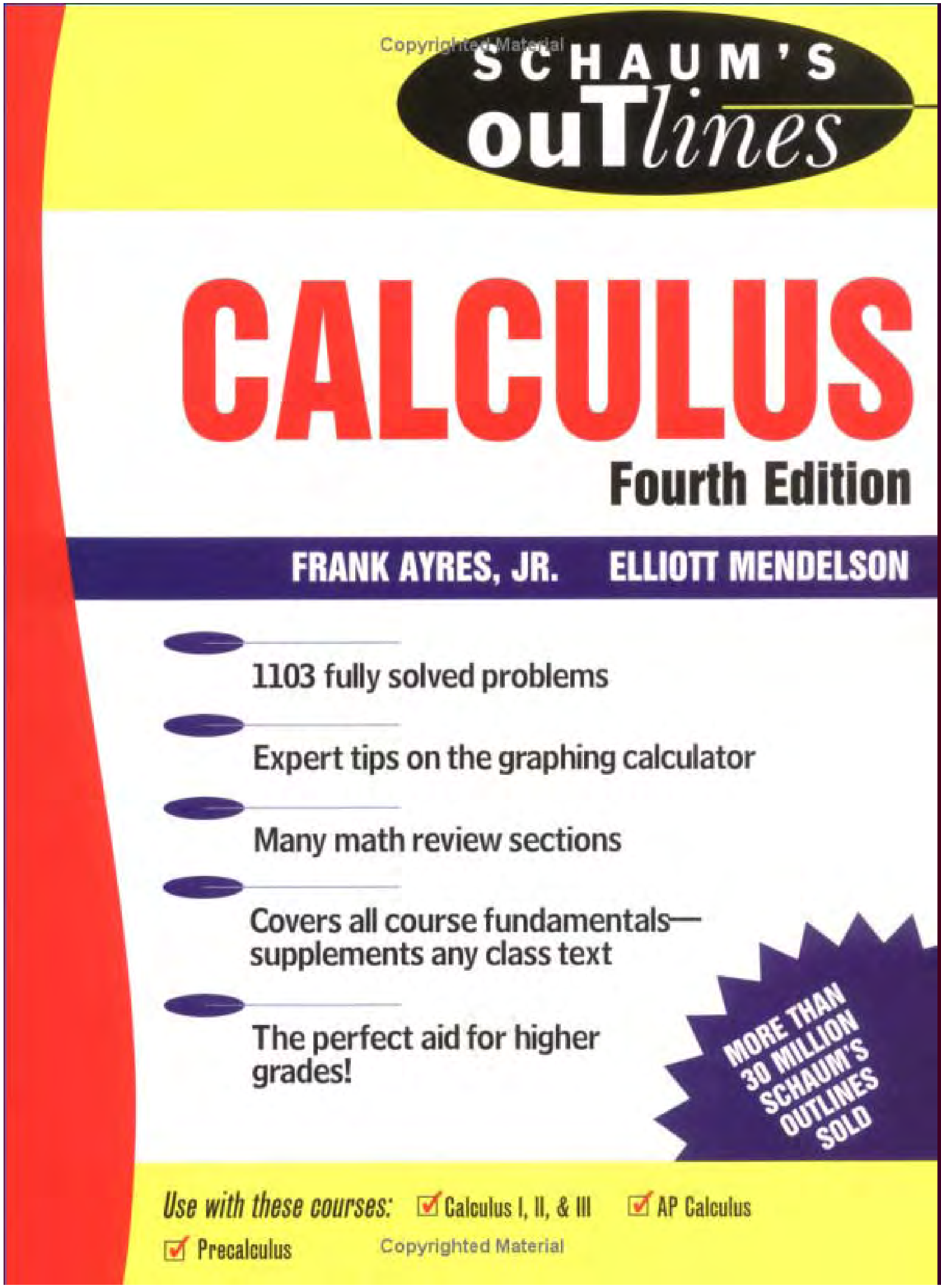
- 1103 fully solved problems
- Expert tips on the graphing calculator
- Many math review sections
- Covers all course fundamentals—supplements any class text
- The perfect aid for higher grades!

Use with these courses: ☑ Calculus I, II, & III ☑ AP Calculus ☑ Precalculus

Copyrighted Material

I Tried This With Commercial Students

- *Many (often, most) of them needed a lot of practice under changing circumstances*
- *But the perceived slow pace of the course made them anxious*
- *And the shorter topic checklist created a marketing disadvantage for my courses.*



SCHAUM'S outlines

CALCULUS

Fourth Edition

FRANK AYRES, JR. ELLIOTT MENDELSON

- 1103 fully solved problems
- Expert tips on the graphing calculator
- Many math review sections
- Covers all course fundamentals—supplements any class text
- The perfect aid for higher grades!

MORE THAN 30 MILLION SCHAUM'S OUTLINES SOLD

Use with these courses: ☑ Calculus I, II, & III ☑ AP Calculus ☑ Precalculus

Back to that Brilliant (?) idea

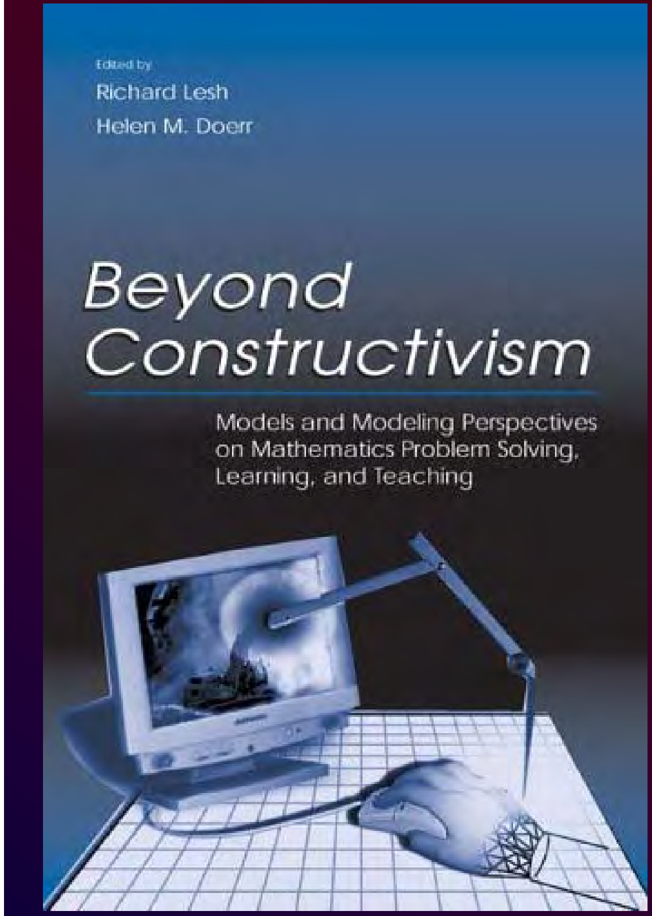
- *Lots of practice exercises*
- *Like we used to do as math students*

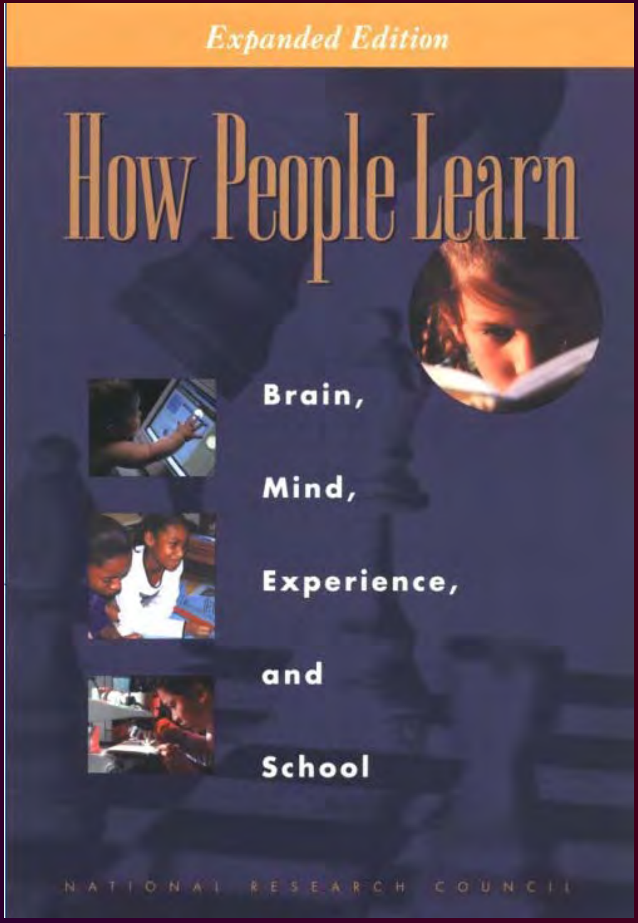
- *It was impractical in commercial training*
- *Now, at last, we can try it on university students.*

Padmanabhan's Thesis: Practice on Domain Testing

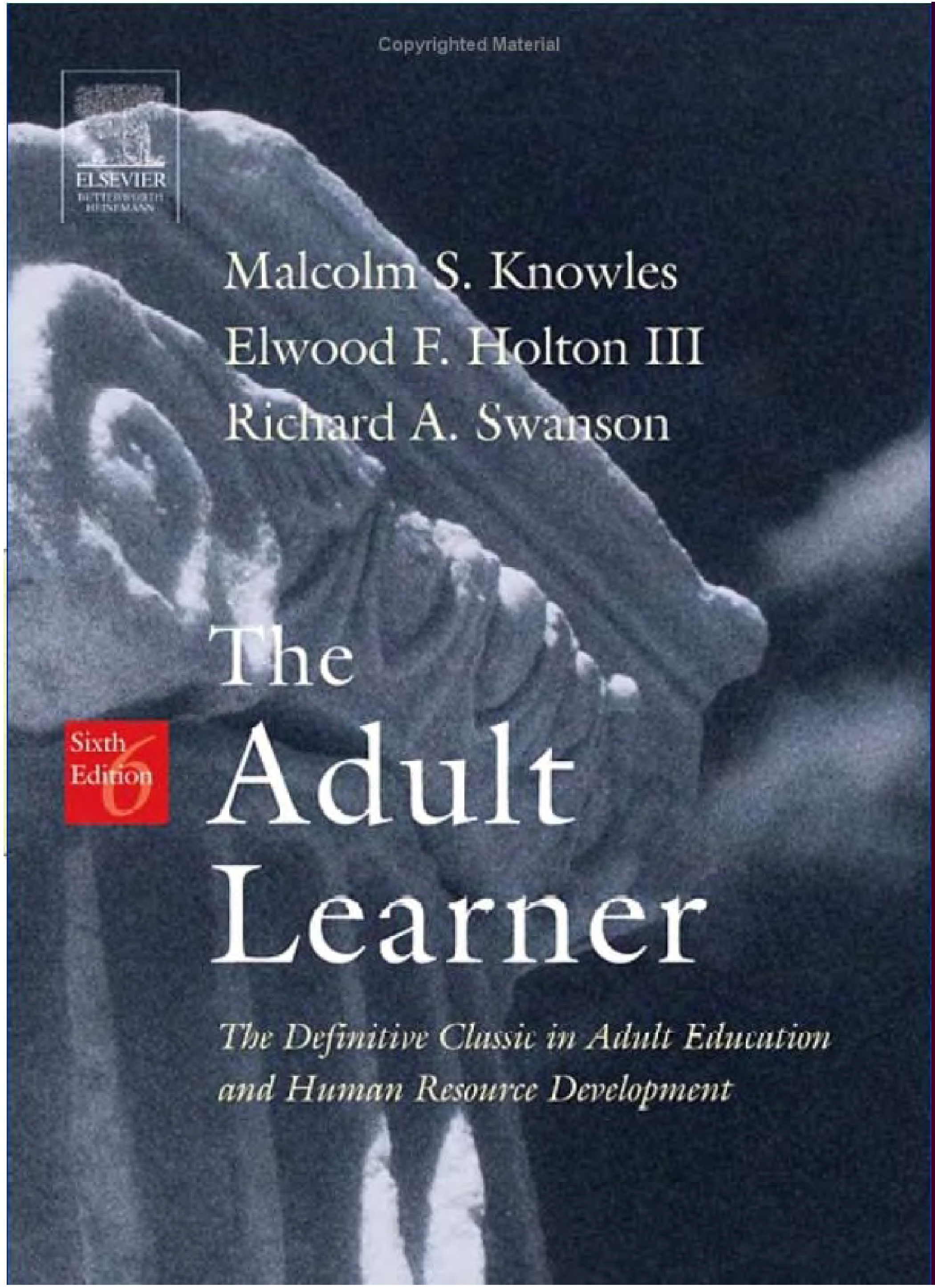
- *15 classroom hours of lecture plus examples plus practice, practice, practice. Lots of procedural instruction and drill*
- *Students mastered every procedure*
- *Final exam*
  - *Applied what they knew to similar questions (near transfer)*
    - *They aced them*
  - *Applied what they knew to a problem that was beyond their practice (not beyond the lecture) (a little bit farther transfer)*
    - *They all failed miserably*
- *Successful transfer of learning requires more than procedural training and practice (This is a what-else-is-new result in science education.)*

Dealing With the Transfer Problem


Edited by
Richard Lesh
Helen M. Doerr
Beyond Constructivism
Models and Modeling Perspectives on Mathematics Problem Solving, Learning, and Teaching



- In science / math education, the transfer problem is driving fundamental change in the classroom
- Students learn (and transfer) better when they discover concepts, rather than by being told them

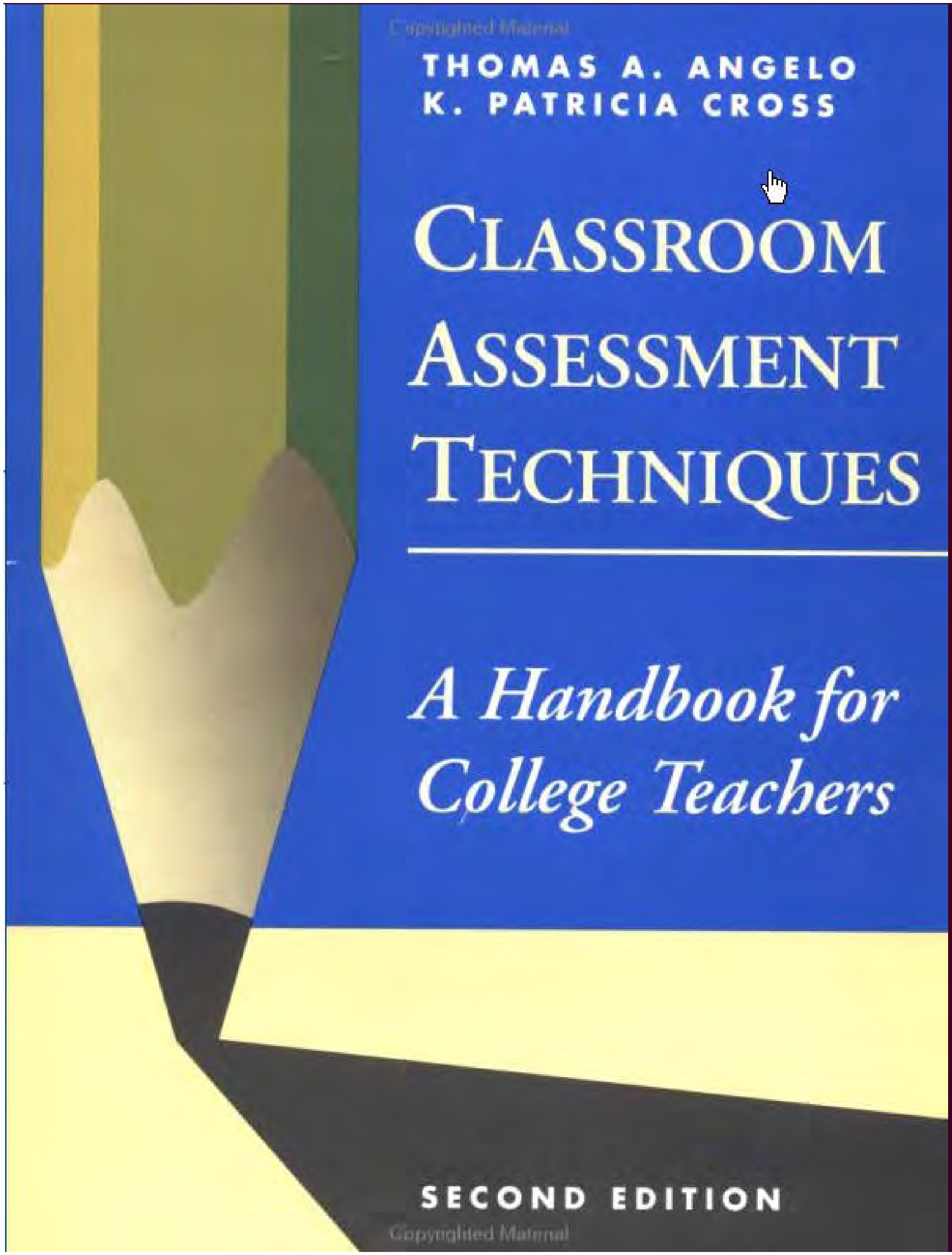


Andragogy

- Pedagogy: study of teaching / learning of children
- Andragogy: study of teaching / learning of adults
- University undergrads are in a middle ground between the teacher-directed child and the fully-self-directed adult
- Both groups, but especially adults, benefit from activity-based and discovery-based styles

Overview

1. *Tour of the Moodle course management system*
2. *Tour of the Black Box Software Testing Course on Moodle*
3. *Overview of use of material like this in the workplace*
4. *Dealing with the instructional challenges of a cognitively complex field of study*
5. ***Taking control of your learning objectives for the course***
6. *Examples of activity patterns*



THOMAS A. ANGELO
K. PATRICIA CROSS

CLASSROOM ASSESSMENT TECHNIQUES

A Handbook for College Teachers

SECOND EDITION

Teaching testing

- *Cognitively complex material*
- *We need to develop skill, judgment, and attitudes, not just knowledge of facts and definitions*
- *We face the usual (for science education) transfer problems*
- *Set a few explicit learning objectives*
- *And assess against them*

Teaching Goals Inventory

- *Take the inventory*
- *If you finish early, start brainstorming an answer to the following questions:*
  - *What courses should testers "have to" take?*
  - *For each course:*
    - *What are its key objectives?*
    - *What is its relevance to software testing?*
- *Over the course of the day, please post your answers to these to the flipcharts*

Teaching Goals Inventory

It is useful to prioritize among goals that fit within 6 categories

- *Discipline-specific knowledge and skill*

  e.g. Develop skill in using materials, tools, technology central to this subject

- *Basic academic success skills*

  e.g. develop listening, reading, and speaking skills; develop appropriate study skills, strategies & habits

- *Higher order thinking skills*

  e.g. develop problem-solving skills

- *Liberal arts and academic values*

  e.g. develop an openness to new ideas; develop an informed historical perspective

- *Work and career preparation*

  e.g. develop ability to work productively with others; improve ability to organize and use time effectively.

- *Personal development*

  e.g. develop a sense of responsibility for one's own behavior

See Angelo & Cross, Classroom Assessment Techniques, 1993.

Teaching Goals Inventory

Cluster	Goals in cluster	Percent rated “essential”	Mean rating
1. Higher order thinking skills	1 – 8 (8)		
2. Basic academic success skills	9 – 17 (9)		
3. Discipline-specific knowledge & skills	18 – 25 (8)		
4. Liberal arts & academic values	26 – 35 (10)		
5. Work & career preparation	36 – 43 (8)		
6. Personal development	44 – 52 (9)		

Teaching Goals Inventory

- *How many did you rate essential?*
- *How are you going to find time to cover all of those?*
- *How well do you have to teach them to cover them well enough?*

- *Commercial training appears to cover massive amounts of material. Unfortunately, we are confounding quantity with quality.*

My Learning Objectives

- *Learn many test techniques well enough to know how, when, and why to use them*
- *Foster strategic thinking--prioritization, designing tests/reports for specific audiences , assess the requirements for complex testing tasks (such as test automation, test documentation)*
- *Apply (and further develop) communication skills (e.g. for bug reporting, status reporting, specification analysis)*
- *Improve and apply teamwork skills (peer reviews, paired testing, shared analysis of challenging problems)*
- *Gain (and document) experiences that can improve the student's chances of getting a job in testing*

Your Learning Objectives

- *Group Discussion*

Overview

1. *Tour of the Moodle course management system*
2. *Tour of the Black Box Software Testing Course on Moodle*
3. *Overview of use of material like this in the workplace*
4. *Dealing with the instructional challenges of a cognitively complex field of study*
5. *Taking control of your learning objectives for the course*
6. ***Examples of activity patterns***