12-19-2018

The Principles of Course Design in Upgrading College Computer Science Courses with Applications to PROG37721

Alexander Tetervak
Sheridan College

Follow this and additional works at: https://source.sheridancollege.ca/teachinglearninginnovations_tla3

Part of the Computer Sciences Commons

SOURCE Citation
Tetervak, Alexander, "The Principles of Course Design in Upgrading College Computer Science Courses with Applications to PROG37721" (2018). Teaching & Learning Academy (TLA3) Final Project. 4.
https://source.sheridancollege.ca/teachinglearninginnovations_tla3/4

Creative Commons License
This work is licensed under a Creative Commons Attribution 4.0 License.
This Report is brought to you for free and open access by SOURCE: Sheridan Scholarly Output, Research, and Creative Excellence. It has been accepted for inclusion in Teaching & Learning Academy (TLA3) Final Project by an authorized administrator of SOURCE: Sheridan Scholarly Output, Research, and Creative Excellence. For more information, please contact source@sheridancollege.ca.
The Principles of Course Design in Upgrading College Computer Science Courses with Applications to PROG37721

Alexander Tetervak B.Ed., Ph.D.
The Question of Interest

• Is it possible to build a framework, based on pedagogical research, for developing and upgrading college Computer Science courses?
  • Determine what can be done
  • Outline a sustainable framework, as much as possible
  • Validate the framework with a representative case of a course upgrade
  • Prepare materials for the consideration phase of the course upgrade
• Developing a framework for course upgrades, as the project, was suggested by Philip Stubbs
Updates versus Upgrades

• Updates
  • Gradual changes, such as improvements of quality, each time when a course is offered
  • Do not require any centralized approach, such as synchronization between sections
  • Do not have any significant impact on other courses of the programs
  • Do not require introducing any changes in the course outlines
  • Workflows of the course updates are straightforward, and usually involve just one step

• Upgrades
  • Require changes in the course outlines, may have a significant impact on other courses
  • The workflows of the upgrades are complicated and involve multiple steps
  • The study is focused on the upgrades
Course Outlines

• Designing a course is not just designing its course outline
• Equivalently, upgrading the course is not equal to upgrading the course outline
• Upgrading the course outline is a separate (or parallel) process
• The language and the structure of the course outline, as well as its workflow, can be different due to various reasons, such as, politics, legal reasons, flexibility, external regulations, etc.
• The result of the development / upgrade of the course outline is posted on the Sheridan’s website, available to everyone
• The results of the course development are posted in the commons sections on SLATE for the teachers, the program coordinators, and the administration. Unlike the course outlines, they are not directly available to the students and general public
Phases of Course Upgrades

Model for Sustainable Curriculum in Applied Computing
AC Faculty VC, SLATE, October 2018
Some specifics of many Computer Science Courses

- Overwhelming amounts of rapidly changing information
  - Challenging to navigate, prioritise, and select (for students, and even for teachers)
  - The teachers have to learn the content continuously
- Close relations and serious dependencies between courses
- The same course is in different programs
- Large numbers of sections and teachers
- Often taught by part-time teachers who have very limited available preparation time
- Relatively high cheating rates
The Framework Validation Course

- PROG37721 / Web services using .NET & C# programming
- Rapidly changing, overwhelming amounts of information
- Needs to be upgraded due to recent drastic changes in the Microsoft technologies
- This course is used in 4 programs
- Large number of sections
  - 679 students, 22 sections, 14 teachers (totals of 2018)
- Prerequisite to a closely-related course
  - PROG35142 / Advanced .NET Server Development
  - Course dependencies and overlaps
  - 88 students, 4 sections, 3 teachers (totals of 2018)
- For many students, about 87% of the students who take the course, it is the first and the last course on Microsoft .NET Technologies
The Developed Framework

- Workflow pattern framework
- Based on the backward course design principle
- Centered on learning outcomes
- Focused on Computer Science Courses
## Backward Design Principle

<table>
<thead>
<tr>
<th>Stage 1</th>
<th>Desired Results</th>
<th>What do we want to achieve?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stage 2</td>
<td>Assessment Evidence</td>
<td>How would we know that we achieved it?</td>
</tr>
<tr>
<td>Stage 3</td>
<td>Learning Plan</td>
<td>How to achieve it?</td>
</tr>
</tbody>
</table>

Understanding by Design is a collection of design patterns for implementations of Backward Design Principle
## The Structure of the Design Stages

<table>
<thead>
<tr>
<th>Stage 1</th>
<th>Desired Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Understandings</td>
<td>Essential Questions</td>
</tr>
<tr>
<td>Knowledge</td>
<td>Skills</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Stage 2</th>
<th>Assessment Evidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance Tasks</td>
<td>Other Evidence</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Stage 3</th>
<th>Learning Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learning Activities</td>
<td></td>
</tr>
</tbody>
</table>
The Upgrade Workflow Scheme

Clockwise iterative workflow, where each petal of the flower is related to its immediate neighbors via the center of the flower.
The Market Value of Courses

Drastic course upgrades are usually caused by changing marked values of knowledge and skills, potentially gained by students in the courses

• PROG38448 / Mobile Java Application Development
  • Blackberry was replaced by Android in the course due to the market change

• PROG36859 / Object Oriented Programming 2
  • Swing was replaced by JavaFX (simpler to use, and more powerful technology) when Sun Microsystems made clear that Swing is a discontinued technology, and new programs should be written using JavaFX
Windows Forms versus WPF

• Job searches using ca.indeed.com
• Local jobs in Toronto, displayed by the site on October 26, 2018

<table>
<thead>
<tr>
<th>Search</th>
<th>Hit Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Windows Forms&quot;</td>
<td>14</td>
</tr>
<tr>
<td>WPF</td>
<td>78</td>
</tr>
<tr>
<td>&quot;Windows Forms&quot; -WPF</td>
<td>5</td>
</tr>
<tr>
<td>WPF -&quot;Windows Forms&quot;</td>
<td>69</td>
</tr>
<tr>
<td>WPF +&quot;Windows Forms&quot;</td>
<td>9</td>
</tr>
</tbody>
</table>

• Windows Presentation Foundation (WPF) is the certain winner
WPF versus UWP

- Job searches using ca.indeed.com
- Local jobs in Toronto, displayed by the site on October 26, 2018

<table>
<thead>
<tr>
<th>Search</th>
<th>Hit Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>WPF</td>
<td>78</td>
</tr>
<tr>
<td>UWP</td>
<td>4</td>
</tr>
<tr>
<td>WPF - UWP</td>
<td>76</td>
</tr>
<tr>
<td>UWP - WPF</td>
<td>2</td>
</tr>
<tr>
<td>WPF + UWP</td>
<td>2</td>
</tr>
</tbody>
</table>

- The current market value of knowledge and skills in Universal Windows Platform (UWP) is still extremely very low, but the situation can change sometime in the future.
Web Forms versus ASP.NET MVC

• Job searches using ca.indeed.com
• Local jobs in Toronto, displayed by the site on October 26, 2018

<table>
<thead>
<tr>
<th>Search</th>
<th>Hit Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Web Forms&quot;</td>
<td>43</td>
</tr>
<tr>
<td>ASP.NET +MVC</td>
<td>326</td>
</tr>
<tr>
<td>&quot;Web Forms&quot; -MVC</td>
<td>18</td>
</tr>
<tr>
<td>ASP.NET +MVC -&quot;Web Forms&quot;</td>
<td>301</td>
</tr>
<tr>
<td>ASP.NET +MVC +&quot;Web Forms&quot;</td>
<td>25</td>
</tr>
</tbody>
</table>

• ASP.NET MVC is the certain winner
ADO.NET Entity Framework

- The Entity Framework is a layer, built on top of ADO.NET, designed to simplify software development.
- Discussing ADO.NET is probably beneficial.
- Training the students how to use the lower level API (ADO.NET) directly, where they have to code tediously to re-implement some features of the already exiting framework, instead of teaching how to use the framework, is one of the serious time-wasters in the current course.
ASP.NET Web API

- The Web API is designed to drastically simply development of Web Services (REST)
- Discussing Windows Communication Foundation (WCF) is probably beneficial
- Training the students in using the lower level API of WCF, for building Web Services (REST) is a ridiculously huge time-waster.
- Functionalities that can be achieved with just a few clicks in Visual Studio, using Web API, require days of tedious coding with WCF directly
- The current situation is so frustrating, that the teachers reduce Web Services to one or two lessons (according to private discussions)
The Market Value of PROG37721

- PROG37721 contains two large parts: Windows Forms and Web Forms, taking together about 75% of the course time.
- Replacing Windows Forms and Web Forms with the newer, simpler and more powerful technologies WPF and ASP.NET MVC seriously increasing the market value of the knowledge and skills gained in the course.
- The estimated impact of the replacement is the increase of the value \(\frac{69 + 301}{5 + 18} = \frac{370}{23} = 16\)\ times in the terms of the job search hits.
The Course Teaching Context

- The Course Teaching Context is an umbrella term, defined and used in the study, to denote significant external factors, such as:
  - The course context in the college programs
  - Expected course delivery (in-class, inverted, online)
  - The available teaching time (the number and the length of the lessons)
  - Other relevant information outside of the course content, for example:
    - Typical student background level of knowledge and skills
    - Usual number of sections
    - Average number of students per section
    - Current number of the teachers
    - Typical levels of the teachers
    - Usual number of part-time teachers (limited time to prepare)
    - If the course is offered in summer terms (two teachers)

- The Course Teaching Context acts as the major filter of the workflow model
The Teaching Context of PROG37721

• For 87% of the students, who take the course, it is the first and the last course on Microsoft .NET technologies

• The course cannot be just a necessary background for the 13% of the students who take the following course. This course must be useful alone, without the following course

• 6 hours per week = about 25 lessons (3 hours each)

• The students already have extensive experience in Java (a very similar language)
  • No need to reteach the students how to program
  • They need to learn the essence of the technologies

• 679 students, 22 sections, 14 teachers (totals of 2018)
The Course Concept

• The brief definition of the essence of the course
• Relates to the course name and the description of the course in the Course Outline
• In this framework, it works as the defined guideline, and also as the coherence check for the course
The Course Concept in the Scheme

Learning Outcomes

Teaching Context

Module Scopes

Course Concept

Resources

Assessments

Market Value
The Course Concept Workflow

- Information on market value of knowledge and skills, potentially gained in the upgraded course, is filtered with the teaching context, reaching current learning outcomes.
- It could be necessary to change the learning outcomes.
- The possible changes in the learning outcomes, filtered with the teaching context in the programs, impact the current course concept.
The Concept of PROG37721

Practically-useful hands-on introduction to current major Microsoft .NET technologies:

- Introduction to .NET and C#
- Windows Presentation Foundation
- ADO.NET Entity Framework
- ASP.NET MVC Framework
- ASP.NET Web API
The Name of PROG37721

• The current name of the course
  Web Services using .NET & C# programming

• The suggested new name
  Microsoft .NET programming using C#

• The rationales for the change
  • Web Service take a very small part in the current version of the course (just 1 or 2 weeks)
  • It is not possible to make Web Services a major part of the coming version of the course
  • The suggested name reflects the course concept
The Updated Course Description

Students are introduced to the current major Microsoft.NET technologies and trained in using these technologies to develop Windows applications, as well as dynamic, data-driven Web applications and Web Services. This course is delivered in two distinct, equal, segments: (1) introduction to .NET and Windows application development, and (2) web application and web services development.
The Updated Learning Outcomes

1. Utilize the C# language and the Framework Class Library effectively to develop .NET applications.
2. Create Windows applications utilizing GUI controls, .NET collections, and file processing.
3. Design and develop data-driven Windows applications using ADO.NET.
4. Create data-driven ASP.NET applications using ADO.NET.
5. Develop Web Services that can be utilized by other .NET applications.
7. Design efficient Graphical User Interfaces (GUIs) for Windows and ASP.NET applications.
8. Debug Windows, ASP.NET and Web Services applications.
Teaching and Learning Resources

• Suitable books and textbooks
• Online tutorials
• Reference resources
• Programming examples
• Video lessons and video tutorials
• Related articles and online presentations
The Resources in the Scheme
The Workflow of the Resources

- The course concept defines criteria for teaching and learning resources, via the teaching context and the learning outcomes.
- The relation is bidirectional: availability of suitable resources drastically impacts concepts of upgrades.
Suitable Textbooks

• **Primary Textbook**
  B. Perkins, J.V. Hammer, and J.D. Raid
  *Beginning C# 7 Programming with Visual Studio 2017*
  Wrox, John Wiley and Sons, 2018, [Link](#)

• **Secondary Textbook**
  A. Troelsen and P. Japikse

• **Supplementary Textbook**
  B. Johnson
  *Professional Visual Studio 2017*
  Wrox, John Wiley and Sons, 2018, [Link](#)

• Each of these textbooks is current and available to the students online via Sheridan Library
The Primary Textbook

- The primary resource
  - Introduction to .NET and C#
  - Windows Presentation Foundation
  - ADO.NET Entity Framework

- The background resource
  - ASP.NET MVC Framework
  - ASP.NET Web API

[Link]
The Secondary Textbook

- The primary resource
  - ASP.NET MVC Framework
  - ASP.NET Web API
- The secondary resource
  - Introduction to .NET and C#
  - Windows Presentation Foundation
  - ADO.NET Entity Framework
- Link
The Supplementary Textbook

- Supplementary resource, on using Visual Studio 2017, for all parts of the course

[Link]
Overlaps with PROG35142

• The upgraded version of PROG37721 is overlapping with PROG35142 / Advanced .NET Server Development in
  • ADO.NET Entity Framework
  • ASP.NET MVC Framework
  • ASP.NET Web API Framework
• The scopes of the overlaps are limited by the chapters of the textbooks (one chapter for each framework)
• There are separate textbooks written on each of these frameworks. Huge amounts of information / activities are still left unexplored for PROG35142
• PROG35145 will definitely benefit from the previous background of the students in these areas
• Naturally, PROG35145 should be adjusted to reflect the upgrade of PROG37721
The Teaching Scopes of Modules

• The Teaching Scopes of Modules (Module Scopes) clarify the module content with redefining the content in terms of assessable statements, and specify approximate numbers of lessons

• Although Modules Scopes use the same language as Learning Outcomes, the scopes are not just Learning Outcomes of the modules
  • Learning Outcomes would define what is required to be in the modules
  • Modules Scopes define what is in the modules, they are exclusive (if it is not mentioned in the scopes, it should not be included)

• Besides the clarification, defining the scopes is also an additional attempt of course standardization
The Assessable Statements

<table>
<thead>
<tr>
<th>Statement</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequential and random access files</td>
<td>Bad</td>
</tr>
<tr>
<td>Discuss implementations of sequential and random access files</td>
<td>OK</td>
</tr>
<tr>
<td>Develop applications processing sequential files</td>
<td></td>
</tr>
<tr>
<td>Discuss implementations of sequential and random access files</td>
<td>Good</td>
</tr>
<tr>
<td>Develop WPF applications loading and saving XML files</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** Implementing random access files from scratch is one of significant time-wasters in the current course. If random access files are required, one should use an imbedded database (library) to implement the feature quicker and better.
The Module Scopes in the Scheme
The Workflow of Module Scopes

- Selected suitable teaching and learning resources, refined by the course context and the learning outcomes, impact the teaching scopes of the modules
- Modules Scopes – Module contents redefined using assessable statements
The Scopes of the Modules – 7A

• Module 1 Scope = 4 lessons
  • Discuss Microsoft .NET platform
  • Master C# background for the rest of the course

• Module 2 Scope = 4 lessons
  • Discuss Windows Presentation Foundation (WPF)
  • Develop desktop applications using WPF

• Module 3 Scope = 5 lessons
  • Discuss ADO.NET
  • Develop data-driven applications using ADO.NET Entity Framework
The Scopes of the Modules – 7B

• Module 4 Scope = 6 lessons
  • Discuss Active Server Pages technology (ASP.NET)
  • Develop data-driven web applications using ASP.NET MVC Framework

• Module 5 Scope = 6 lessons
  • Discuss Web Services
  • Discuss Windows Communication Foundation (WCF)
  • Develop Web Services using ASP.NET Web API
  • Develop web applications consuming Web Services

The amounts of programming may look overwhelming for the course, but it is not the case: large parts of the applications are simply generated with a few clicks in Visual Studio.
The Course Materials

• Selected parts of the course resources
  • Specific chapters of the textbooks
  • Parts of online tutorials
  • Sections of API documentation

• Lesson Slides
  • Emphasise major points
  • Guide through the resources
  • Outline lesson flow

• Programming examples
  • Fixing and updating textbook code is unavoidable (it stops working after library updates)

• Activities

• Self Assessments
The Course Materials in the Scheme
The Workflow of Course Materials

1. The module scopes guide development of course materials
2. The development of the course materials is filtered by the teaching context and the learning outcomes
3. The relation is bidirectional: the course materials clarify the module scopes
The Assessments in the Scheme
The Workflow of Assessments

- The course materials define assessments, refined by the course context and the learning outcomes.
- The relation is bidirectional:
  - Assessments impact course materials and module structure of the course.
  - The materials are developed with potential assessments in mind (backward design).
Updated Evaluation for PROG37721

• Evaluation – 7A = 50%
  • 2 Exercises (Labs) x 5% each = 10%
  • 1 Assignment = 10%
  • 1 Quiz = 5%
  • Midterm Exam = 25%
    • Part1 (pencil and paper) = 15%
    • Part 2 (laptop and open books) = 10%

• Evaluation – 7B = 50%
  • 2 Exercises (Labs) x 5% each = 10%
  • 1 Assignment = 10%
  • 1 Quiz = 5%
  • Midterm Exam = 25%
    • Part1 (pencil and paper) = 15%
    • Part 2 (laptop and open books) = 10%
Conclusions

• The study defined a general workflow framework, based on the Backward Design Principle, centered on the Learning Outcomes, focused on upgrading Computer Science courses.
• Further refinement of the framework is probably necessary.
• The framework is validated with the case of upgrading the representative course (PROG37721).
• Proceeding with this course upgrade requires further discussions with the colleagues, the program coordinators, and the administration.
References

• Nilson, L. B.
  *Teaching at Its Best: A Research-Based Resource for College Instructors*

• Fink, L. D.
  *Creating Significant Learning Experiences: An Integrated Approach to Designing College Courses*

• McTighe, J.; Wiggins, G.
  *The Understanding by Design Guide to Advanced Concepts in Creating and Reviewing Units*
  Association for Supervision and Curriculum Development (ASCD), 2012

• McTighe, J.; Wiggins, G.
  *The Understanding by Design Guide to Creating High-Quality Units*
  Association for Supervision and Curriculum Development (ASCD), 2011

• Wiggins, G.; McTighe, J.
  *Understanding by Design*
  Association for Supervision and Curriculum Development (ASCD), 2005