Submitted by: Computer Science Department

Department/Discipline(s) and Course Number(s): Computer Science CPSC 350

Course Title: Applications of Databases

Type of change (check all applicable):

Number* _____ Title_____ Credits_4____ Description_X__ Prerequisites _X__ Deletion_____ Cross list** _____

*This course number must be approved by the Office of the Registrar before the proposal is submitted.

**To cross list courses between departments/colleges, there should be two cover sheets submitted with the proposal – one by the chair of each department with signatures from the relevant College Curriculum Committee Chair.

Effective Date: FALL Semester, Year 2015

Current Catalog Entry
CPSC 350: Application of Databases (3)
Prerequisite: CPSC 230
Presents logical database modeling and design, emphasizing the construction and analysis of relational schemas. Covers semantics, integrity constraints, functional dependencies, and table normalization. Practical use of relational algebraic operators, the tuple relational calculus, and their expression in declarative query languages. Introduces basic data mining concepts. May involve student team projects to develop small but representative data collection and analysis applications.

Proposed Catalog Entry
CPSC 350: Application of Databases (4)
Prerequisite: (1) CPSC 225 and Grade of C or better in CPSC 220, or (2) CPSC 340
Presents basic techniques for the design and implementation of database-driven web applications. Topics include the design of relational and NoSQL databases and scaling techniques such as the use of load balancing and distributed systems. Programming intensive using a dynamic high-level general-purpose language.

JUSTIFICATION (including impact on majors, minors, concentrations, and general education courses within the University curriculum; attach additional pages if required)
Additional material added to conform with the curriculum guidelines of the Association for Computing Machinery and the IEEE Computer Society. Proposed Content shown on following page.

**TRANSITION PLAN** *(describe how will students who are in Catalogs where the course is required for a major be accommodated; attach additional pages if required)*

Students who are in catalogs were the 3 credit version of the course is required can take the 4 credit version to fulfill the requirement.

**Approvals**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Jennifer Polack</td>
<td>Date:</td>
</tr>
<tr>
<td>Department Chair</td>
<td></td>
</tr>
<tr>
<td>College Curriculum Chair</td>
<td>Date:</td>
</tr>
</tbody>
</table>

Expedites course changes are posted for a 10-class day comment period. If no comments are raised during that time, the proposal becomes final. All expedited proposals approved in this way will be noted on the UCC web site.

If comments are raised, the proposal may be reviewed by the UCC and then approved or it may be returned to the CCC for additional deliberation (as required).

**Proposed content (from ACM/ IEEE guidelines)**

<table>
<thead>
<tr>
<th>area</th>
<th>topics covered</th>
<th>hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Information Management Concepts (ACM core topic area — ACM recommends 3 core hours)</td>
<td>Basic information storage and retrieval concepts. Information management applications Quality issues.</td>
<td>1</td>
</tr>
<tr>
<td>Database Systems (ACM core topic area — ACM recommends 3 core hours)</td>
<td>Components of database systems Use of declarative query language Approaches to database systems</td>
<td>6</td>
</tr>
</tbody>
</table>
| Data Modeling (ACM core topic area— ACM recommends 4 core hours) | Relational data models  
Semi-structured data models | 3 |
| Relational Databases (ACM elective) | Relational database design  
Functional dependency  
Normal forms | 5 |
| Query Languages (ACM elective) | SQL  
Joins  
Aggregates and group by Subqueries | 6 |
| Distributed Databases (ACM elective) | Distributed data storage  
Use of the MapReduce processing model | 4 |
| Approached for managing large volumes of data (ACM elective) | noSQL  
Graph databases (e.g., Neo4J)  
Semi-structured data  
couchDB  
REDIS | 6 |
| Indexing (ACM elective) | Impact of indexes on query performance  
Creating indexes with SQL | 2 |
| Human Computer Interaction | | |
| Programming Interactive Systems (ACM elective) | Model-view controller  
Data driven applications (database backed web pages) | 4 |
| Platform Based Development | | |
| Web Platforms (ACM elective) | Web programming languages.  
Design and implementation of web applications  
Software as a service | 4 |
| Information Assurance and Security | | |
| Defensive Programming (ACM core topic area — ACM recommends 2 core hours) | Input validation and data sanitization  
SQL injection  
XSS vulnerability | 2 |
| Software Engineering | | |
| Software Project Management (ACM core topic area — ACM recommends 2 core hours) | team processes effort estimation scheduling and tracking | 2 |
| TOTAL HOURS | | = SUM(C2:C18) \# "0" \* MERGEFORMAT45 |