Ninth LACCEI Latin American and Caribbean Conference (LACCEI'2011), Engineering for a Smart Planet, Innovation, Information Technology and Computational Tools for Sustainable Development, August 3-5, 2011, Medellín, Colombia.

Hybrid Tools for Infrastructure Construction Business Intelligence

Manuel E. Bermudez
University of Florida, Gainesville, Florida, USA, manuel@cise.ufl.edu

ABSTRACT

We describe a set of tools for intelligently managing infrastrucutre construction projects, developed by a group of undergraduate students at the University of Florida, under a University-Industry Cooperation program known as IPPD (Integrated Product and Process Design), and sponsored by InfoTech, Inc. a Gainesville high-technology company. We describe the nature of the project and the tools that have been produced, and describe the educational program (IPPD) under which they were developed.

Keywords: infrastructure construction, intelligent search tools, data mining

1. INTRODUCTION

The author's IPPD project this year at the University of Florida consisted of helping our sponsor company, InfoTech, Inc. [1], to expand the capabilities of one its primary software products, called BidExpress [2]. Info Tech provides applications to help facilitate construction project pricing and online bidding for state infrastructure projects (roads and bridges) throughout the United States. These bids are receive by, and contracts awarded by, Departments of Transportation (DOT) in each state. The Bid Express application can approximate the prices for specific items needed for a new construction project based on data from previous projects. The system can help contractors by building up an approximate bidding price for their project, allowing them to make more competitive bids on projects, and increasing their chances of being successful in their bids.

Info Tech has a large amount of historical data to approximate these prices accurately, but is currently unable to use the data very efficiently or very intelligently. Construction items have different identification numbers, different abbreviation and naming conventions, and different units across states. For example, guardrail may be abbreviated "GDRL" in Tennessee, but in Georgia it may be abbreviated "GRL". In addition, Georgia specifies amounts of concrete in square meters while Florida specifies amount of concrete in square yards. These differences make it difficult to group the item data to make an accurate cost projection for a specific item in a project bid. BixExpress' search mechanism currently returns an item if the searched string is a substring of the item name or description. A search for too short a substring will return extraneous and excessive results. For example, a search for "concrete" in the Georgia database returns about 11,000 results including "asphaltic concrete" and "concrete curb". A search for "conc" returns over 34,000 results, including a result for "electrical junction box, conc ground mounted." Clearly, results in such overwhelming quantities are not useful to a contractor trying to construct a contract bid.

This year, InfoTech signed up to sponsor an IPPD [3] project at the University of Florida. We describe the IPPD program in some detail in Section 6. The student team, consisting of six students, dubbed themselves "Onyx Intelligence", and as with all IPPD projects, they designed their own logo, shown in Figure 1.

Figure 1. The Onyx Intelligence logo.

Our task was to improve the search functionality, to make it more “intelligent”, and to return, as much as possible, only highly relevant items. Specifically, our key objectives were to design intelligent and implement a “search engine” software system for mining “similar” construction terms from the database, and design and implement a system for creating conversion rules between similar items. The remainder of this paper is describes the design and development process, as it proceeded throughout the year. In Section 2 we discuss the software requirements, including the customer’s prioritized needs. In Section 3 we discuss the system design, and how it was broken down into two sub-systems. In Section 4 we discuss the system’s implementation, and the various software technologies. In Section 5 we show some the the results of the new system. In Section 6 we discuss the IPPD program under which this project was carried. Finally, in Section 7, we present our conclusions.

2. REQUIREMENTS

IPPD projects are carried out over two-semester periods. During the first semester, students study the problem and relevant literature, and develop an initial design and prototype. The second semester is devoted to implementation, testing, and validation. During the first semester (August through December, 2010), the Onyx Intelligence team had the schedule of deliverables shown in Figure 2.

Deliverable	Date
Technical Strategy	9/13/2010
System/Product Requirements	9/13/2010
Complete & Testable Product Specs	9/20/2010
System/Product Architecture	9/20/2010
Subsystem Design	10/6/2010
Preliminary Comprehensive Test Plan	10/6/2010
Preliminary Design Report	10/6/2010
Product Architecture	10/20/2010
Software and Hardware Prototype Plan	10/20/2010
Component Design Specifications	10/27/2010
Analytical & Experimental Plans PowerPoint	11/3/2010
Comprehensive Test Plan PowerPoint	11/10/2010
Name Association	11/29/2010
System Level Design Report Preliminary PowerPoint	11/29/2010
System Level Design Report	12/15/2010

Figure 2. Schedule of deliverables for the first semester.

The students first addressed the customer’s needs. The initial list of needs was as follows:

- The price approximations for construction parts are not accurate enough
- These construction tool abbreviations for my neighbor state are confusing
- How can I keep up with all of the new names for tools?
- What is the item description?
- Can I change the units?
- Can I filter my search results?
- Does the price approximation vary by geographical location within a state?
- Can I change the amount of results I can view on a page?
- Is there an easier way to see history of what people bid previously on these jobs?

The students then classified and prioritized the customer's needs, as shown in Table 1.

	No.	Category	Need	Priority
Product	1	feature	Item Relevancy	5
	2	feature	Back-end User Interface	2
	3	feature	Interstate Searches	5
	4	feature	Automatically Add Item Relationships	5
	5	feature	Search by Date	2
	6	feature	Search by given states	4
	7	feature	Search by unit of measure	4
	8	quality	Abbreviation Connections	5
		quality	Page Result Count	3

Priority	Rating
5	High
4	medium/high
3	Medium
2	medium/low
1	Low

Table 1. Prioritized customer needs.

The next step was to formulate the system requirements, which appear in Table 2.

No.	Requirement Description
1	The database shall consolidate the search results to return the most 100 (speculative figure) related records.
2	The system shall be able to show how the price of an item has changed every year over the past 10 years.
3	The system shall be able to convert units of a quantifiable item into all similar unit types. Ex: LM, LM^1 m, ft, in can all be converted to meters (m).
4	The system shall include a smart search that can add relationships or create similarities between terms after approximately 1000 (speculative figure) searches for each of these terms.
5	The system shall be able to offer a suggested spell check for a construction term misspelled by 1 letter.
6	The system should be able to work in all of the four main browsers. (Safari, Chrome, Firefox, Internet Explorer)
7	The system should be able to support all 128 ASCII symbols.
8	The system shall be able to recognize the " " as well as the AND and OR searching functions when interpreting a search.
9	The system should respond with search results within 5 seconds (speculative figure) of starting the search.
10	The system shall be able to provide a number of potential filtering options by state and by date.
11	Ask the user for feedback regarding the relevancy of the search results.

Table 2. System Requirements.

The next step was to develop a "House of Quality", a mapping between the customer needs and the requirements specification. The customer needs, already rated in order of importance, were classified into three categories (cost, features, and quality), and then rated against the requirements metrics. By multiplying the values in the columns by the importance rating of the customer need, an importance rating can be placed on the specifications. The result is shown in Table 3.

Specification

	Category	Customer Need	Imp.	Filter Results	Price History	Unit Conversion	Smart Search	Spell Check	Browser Compatibility	ASCII Support	Search Logic	Search Speed	Search Filter Options	User Feedback
Product	Cost	Accurate Bid Estimations	3	3		9	9		3			3	1	9
	Features	Abbreviation Connections	4	1			9	1		3	3			
		Geographical Estimates	3	3		9	3	1					9	
		Item Description	2				1	9						1
		Bid History Link	2		9	1			3			1		
	Quality	Advanced Search Options	3	3		3	9	1			3		9	
		Page Result Amount	1	9		1	3		1					
		Intelligent Search	5	1		3	9	1		3	9	3	1	9
		Weighted Column Total		**45**	**18**	**81**	**149**	**33**	**16**	**27**	**66**	**26**	**62**	**74**
		Rank		**6**	**10**	**2**	**1**	**7**	**11**	**8**	**4**	**9**	**5**	**3**

IMPORTANCE RANGE 1 to 5
Correlation: 1=Low, 3=Medium, 9=High

Table 3. "House of Quality", a mapping of customer needs to requirements specifications.

3. SYSTEM DESIGN

In order to improve the search engine, the diagram in Figure 4 displays a set of simpler sub-problems to facilitate the design process. Partial design solutions to these sub-problems were further assessed.

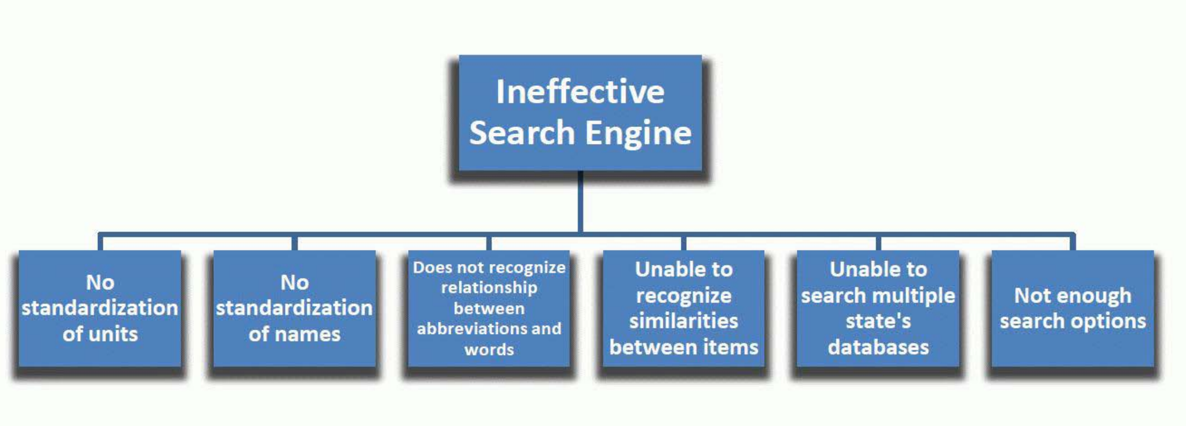
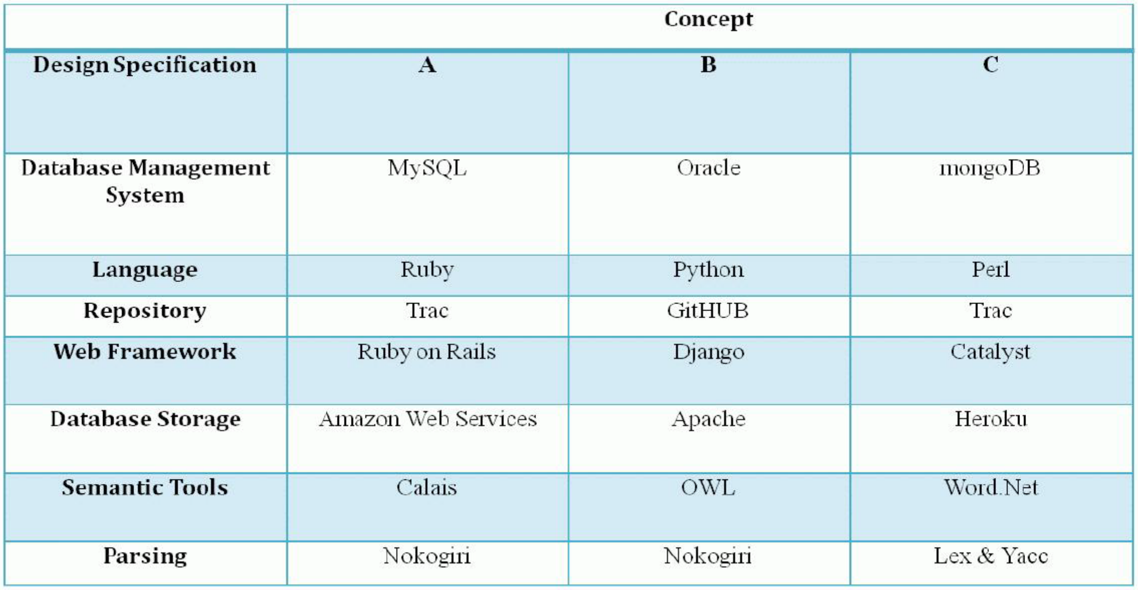


Figure 4. Decomposition of the Problem.

Many design alternatives were considered, for the database management system, for the programming language, the data repository, the web framework, the database storage, semantic tools, and tools for parsing DOT documents. The choices are shown in Figure 5.

Figure 5. Design choices for system components.

Three leading concept combinations were chosen for further examination. These are shown in Figure 6.



	Concept		
Design Specification	**A**	**B**	**C**
Database Management System	MySQL	Oracle	mongoDB
Language	Ruby	Python	Perl
Repository	Trac	GitHUB	Trac
Web Framework	Ruby on Rails	Django	Catalyst
Database Storage	Amazon Web Services	Apache	Heroku
Semantic Tools	Calais	OWL	Word.Net
Parsing	Nokogiri	Nokogiri	Lex & Yacc

Figure 6. Leading combinations of design choices.

A concept scoring matrix was developed, to aid in the final choices of tools. The concept scoring matrix is shown in Figure 7. The highest score was for Concept A, which we adopted for our design.

Selection Criteria	Weight	A	B	C
Compatibility	5%	4	2	3
Storage Capacity	1%	3	4	4
Customer Preference	40%	2	4	1
Accessibility	5%	3	2	3
Familiarity	9%	4	1	2
Support	5%	4	2	3
Cost	35%	3	1	3
Total	100%	2.79	1.965	2.12
Rank		1	3	2
Continue?		Yes	No	No

Figure 7. Concept scoring matrix.

The system architecture was then designed, based on our concept selection. The architectural diagram is shown in Figure 8.

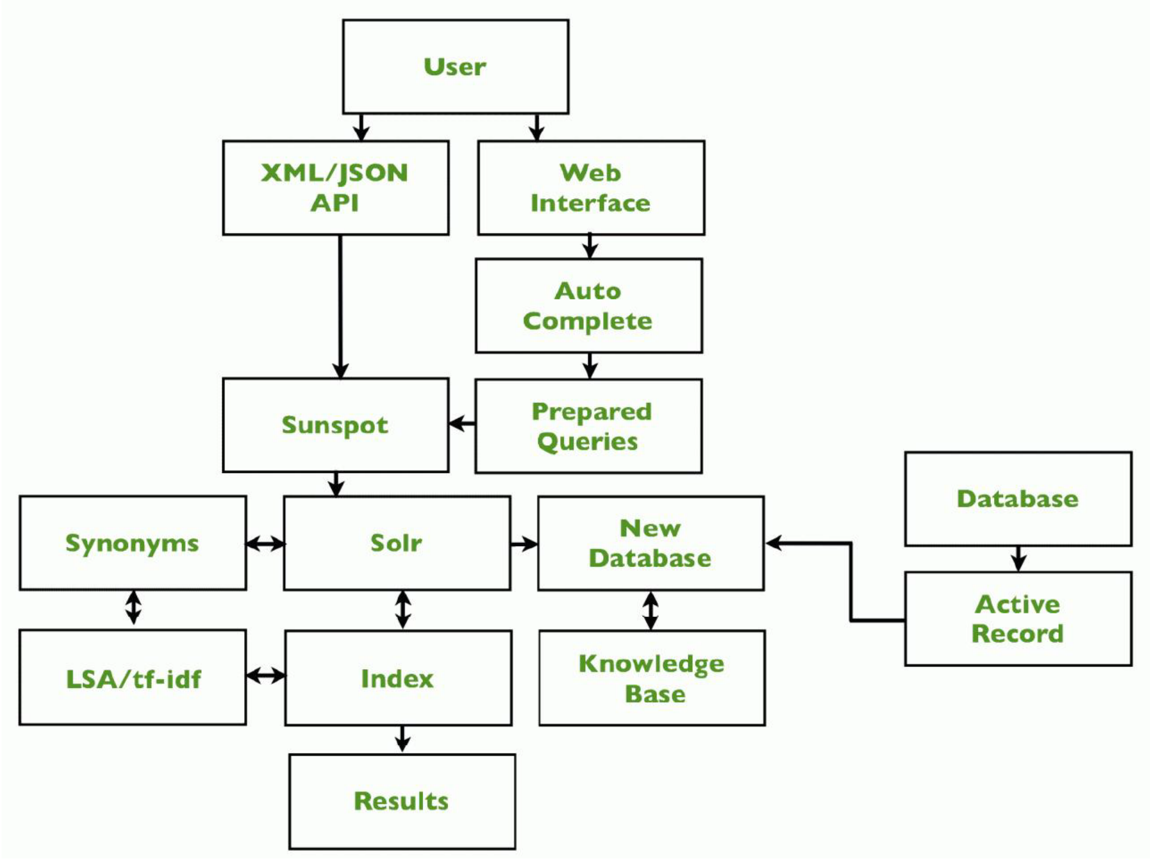


Figure 8. System architecture diagram.

The database schema and interactions were designed next. It appears in Figure 9.

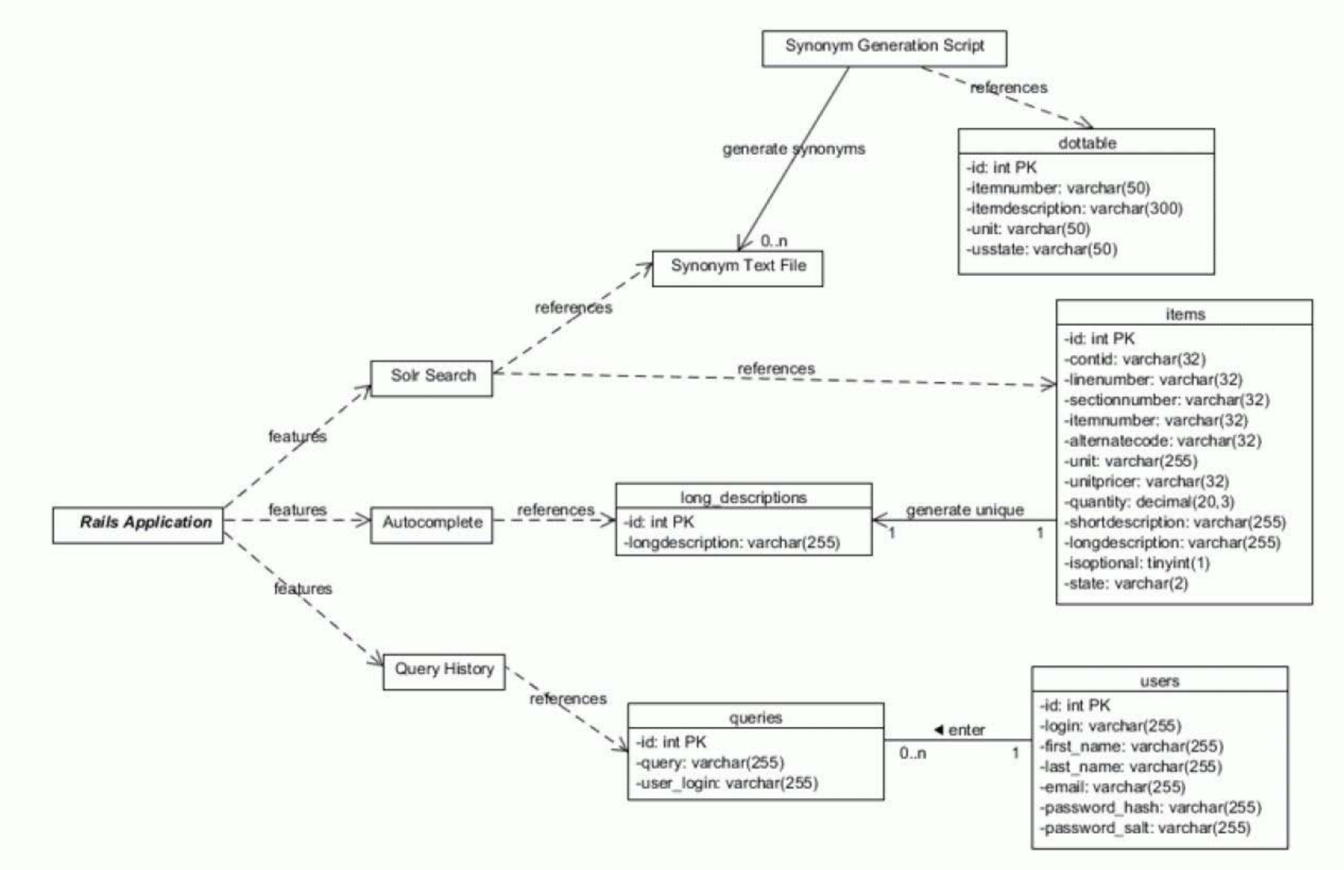


Figure 9. Database schema and interactions.

4. SYSTEM IMPLEMENTATION

The implementation phase was carried out during the second semester of the two-semester project, from January to April this year. A project plan was developed and executed. The project plan is shown in Figure 10.

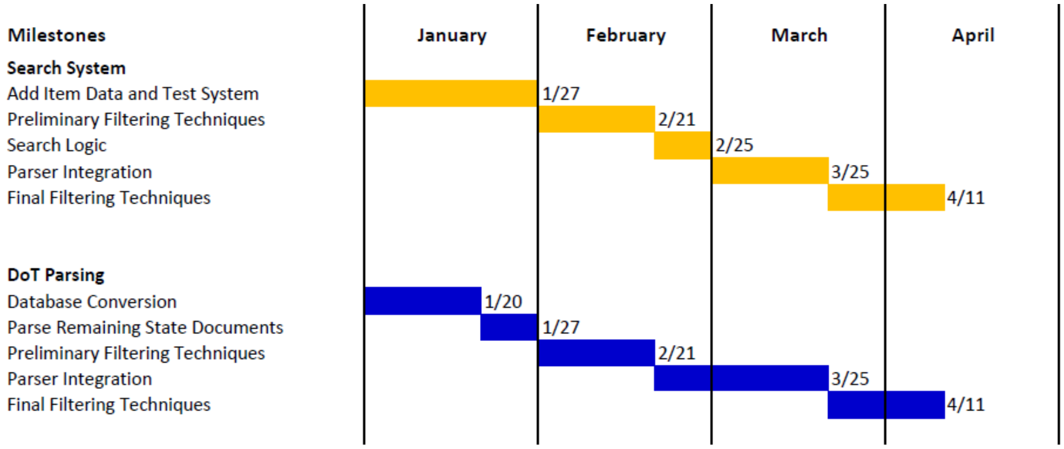


Figure 10. Project plan for the second semester, the implementation phase.

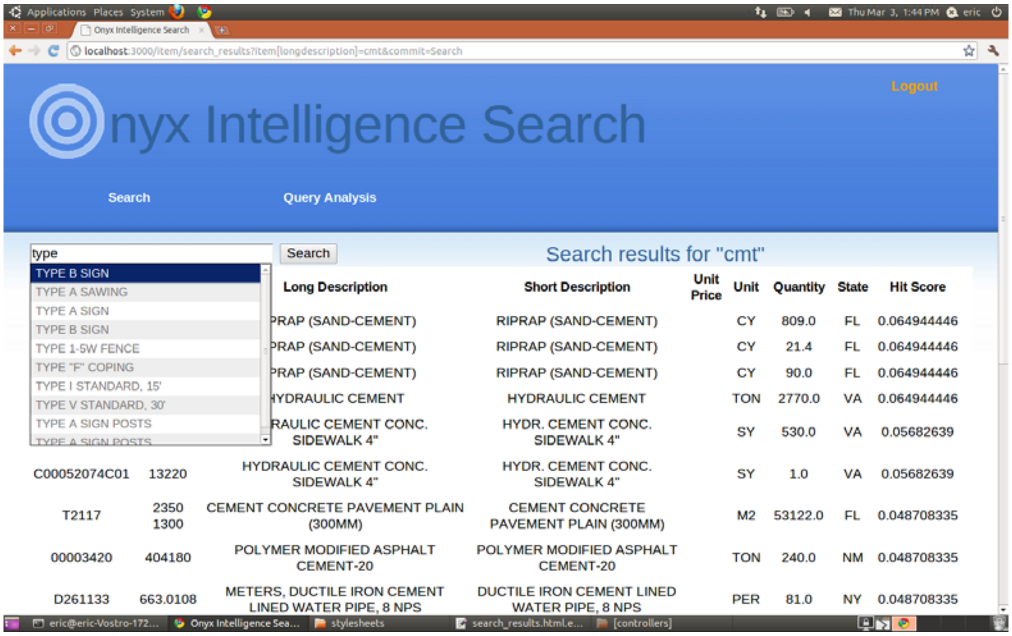
A computer that deploys the application should have the following software installed on it:

- Ruby (version 1.8.7) [4]
- Rails(version 2.3.8) [5]
- Activemodel (version 3.0.1) [6]
- Haml (version 3.0.24) [7]
- Mysql (version 2.8.1) [8]
- Nokogiri (version 1.4.3.1) [9]
- Rake (version 0.8.7) [10]
- Rsolr (version 0.12.1) [11]
- Sunspot rails (version 1.1.0) [12]

The databases are to be hosted on a MySQL server. To run a Solr server: Solr requires Java 1.5 and an application server (such as Tomcat) which supports the Servlet 2.4 standard.

5. SYSTEM RESULTS

Here we show a few screen shots that demonstrate the capabilities of the new, enchanced search engine. Due to intellectual property restrictions (all IP belongs to Info Tech, Inc.) we do not show screen shots from BidExpress, but from the Onyx Intelligence software, for which the students developed a graphical user interface. Figure 11 shows the new auto-complete capability.

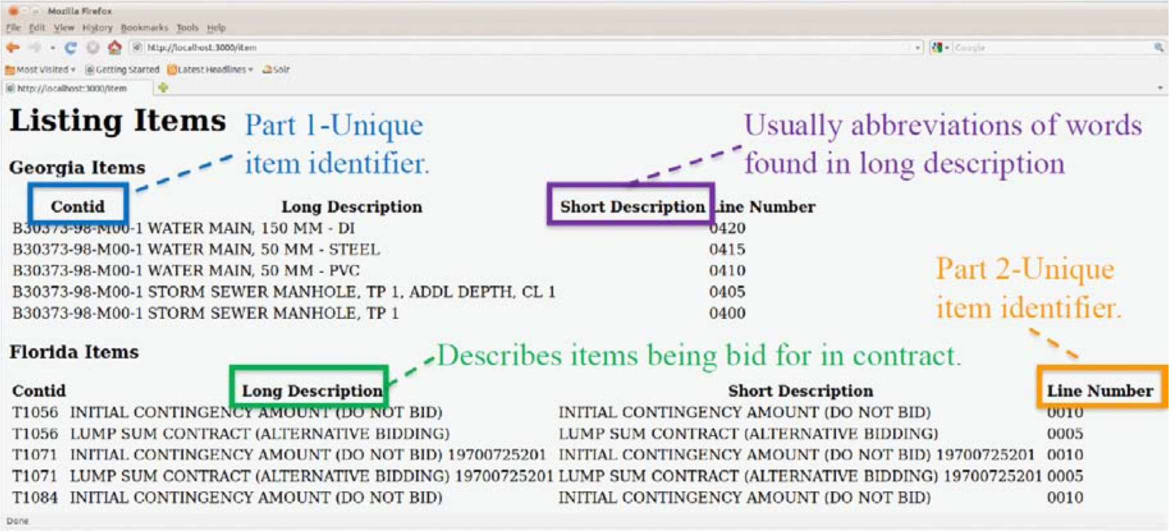


Search results for "cmt"

		Long Description	Short Description	Unit Price	Unit	Quantity	State	Hit Score
		PRAP (SAND-CEMENT)	RIPRAP (SAND-CEMENT)		CY	809.0	FL	0.064944446
		PRAP (SAND-CEMENT)	RIPRAP (SAND-CEMENT)		CY	21.4	FL	0.064944446
		PRAP (SAND-CEMENT)	RIPRAP (SAND-CEMENT)		CY	90.0	FL	0.064944446
		YDRAULIC CEMENT	HYDRAULIC CEMENT		TON	2770.0	VA	0.064944446
		RAULIC CEMENT CONC. SIDEWALK 4"	HYDR. CEMENT CONC. SIDEWALK 4"		SY	530.0	VA	0.05682639
C00052074C01	13220	HYDRAULIC CEMENT CONC. SIDEWALK 4"	HYDR. CEMENT CONC. SIDEWALK 4"		SY	1.0	VA	0.05682639
T2117	2350 1300	CEMENT CONCRETE PAVEMENT PLAIN (300MM)	CEMENT CONCRETE PAVEMENT PLAIN (300MM)		M2	53122.0	FL	0.048708335
00003420	404180	POLYMER MODIFIED ASPHALT CEMENT-20	POLYMER MODIFIED ASPHALT CEMENT-20		TON	240.0	NM	0.048708335
D261133	663.0108	METERS, DUCTILE IRON CEMENT LINED WATER PIPE, 8 NPS	DUCTILE IRON CEMENT LINED WATER PIPE, 8 NPS		PER	81.0	NY	0.048708335

Figure 11. Auto-complete capability.

Figure 12 shows the new multi-state search capability, and the relevant items: the item identifier, short and long descriptions, and the line number in the DOT document.



Listing Items

Part 1-Unique item identifier.

Usually abbreviations of words found in long description

Part 2-Unique item identifier.

Describes items being bid for in contract.

Georgia Items

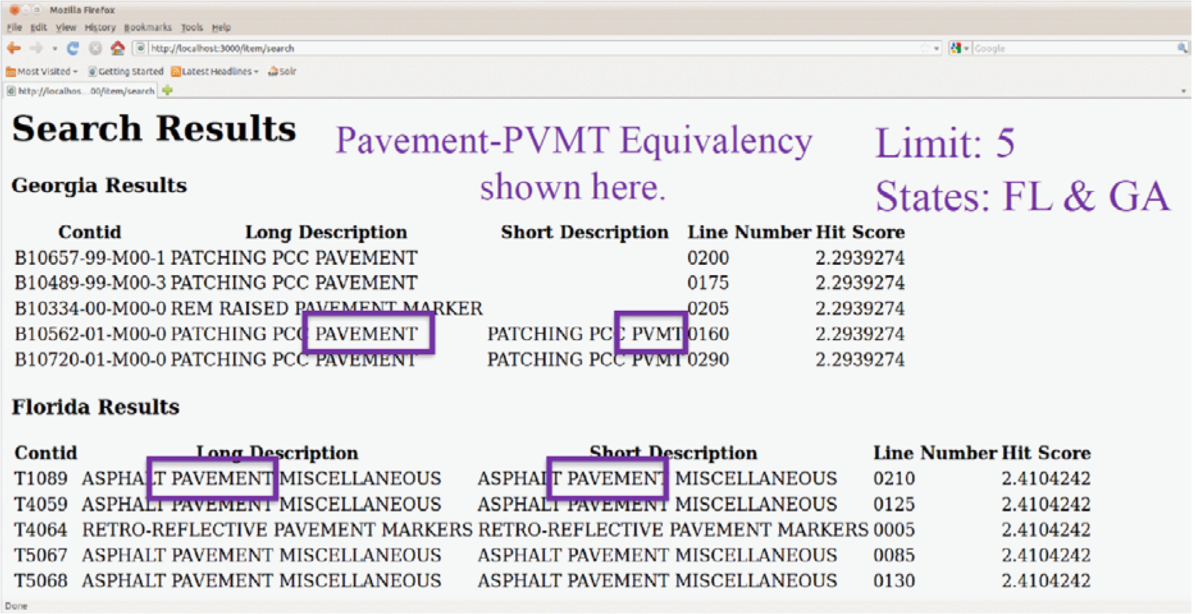
Contid	Long Description	Short Description	Line Number
B30373-98-M00-1	WATER MAIN, 150 MM - DI		0420
B30373-98-M00-1	WATER MAIN, 50 MM - STEEL		0415
B30373-98-M00-1	WATER MAIN, 50 MM - PVC		0410
B30373-98-M00-1	STORM SEWER MANHOLE, TP 1, ADDL DEPTH, CL 1		0405
B30373-98-M00-1	STORM SEWER MANHOLE, TP 1		0400

Florida Items

Contid	Long Description	Short Description	Line Number
T1056	INITIAL CONTINGENCY AMOUNT (DO NOT BID)	INITIAL CONTINGENCY AMOUNT (DO NOT BID)	0010
T1056	LUMP SUM CONTRACT (ALTERNATIVE BIDDING)	LUMP SUM CONTRACT (ALTERNATIVE BIDDING)	0005
T1071	INITIAL CONTINGENCY AMOUNT (DO NOT BID) 19700725201	INITIAL CONTINGENCY AMOUNT (DO NOT BID) 19700725201	0010
T1071	LUMP SUM CONTRACT (ALTERNATIVE BIDDING) 19700725201	LUMP SUM CONTRACT (ALTERNATIVE BIDDING) 19700725201	0005
T1084	INITIAL CONTINGENCY AMOUNT (DO NOT BID)	INITIAL CONTINGENCY AMOUNT (DO NOT BID)	0010

Figure 12. New multi-state capability, with relevant items.

Finally, in Figure 13 we show the new capability of recognizing similar but not identical items, in this case “Pavement”, over multiple states.



Search Results

Pavement-PVMT Equivalency shown here.

Limit: 5
States: FL & GA

Georgia Results

Contid	Long Description	Short Description	Line Number	Hit Score
B10657-99-M00-1	PATCHING PCC PAVEMENT		0200	2.2939274
B10489-99-M00-3	PATCHING PCC PAVEMENT		0175	2.2939274
B10334-00-M00-0	REM RAISED PAVEMENT MARKER		0205	2.2939274
B10562-01-M00-0	PATCHING PCC PAVEMENT	PATCHING PCC PVMT	0160	2.2939274
B10720-01-M00-0	PATCHING PCC PAVEMENT	PATCHING PCC PVMT	0290	2.2939274

Florida Results

Contid	Long Description	Short Description	Line Number	Hit Score
T1089	ASPHALT PAVEMENT MISCELLANEOUS	ASPHALT PAVEMENT MISCELLANEOUS	0210	2.4104242
T4059	ASPHALT PAVEMENT MISCELLANEOUS	ASPHALT PAVEMENT MISCELLANEOUS	0125	2.4104242
T4064	RETRO-REFLECTIVE PAVEMENT MARKERS	RETRO-REFLECTIVE PAVEMENT MARKERS	0005	2.4104242
T5067	ASPHALT PAVEMENT MISCELLANEOUS	ASPHALT PAVEMENT MISCELLANEOUS	0085	2.4104242
T5068	ASPHALT PAVEMENT MISCELLANEOUS	ASPHALT PAVEMENT MISCELLANEOUS	0130	2.4104242

Figure 13. New similar item matching, over multiple states.

6. STUDENT PARTICIPATION.

In this project we had six senior-level undergraduate students, five from Computer Engineering, and one from Industrial Engineering. The author served as faculty coach, and Info Tech, Inc. appointed two liaison engineers. The students were all enrolled in the University of Florida College of Engineering’s Integrated Product and Process Design (IPPD) program. In the IPPD program, an interdisciplinary group of senior-level students, of various engineering disciplines, engage in a product or process design project for an industrial sponsor, under the supervision of a faculty coach and liaison engineers from the sponsoring company. The students spend two semesters in this capstone design experience, designing and building authentic products for those industry sponsors. Participating students are seniors from the College of Business, and all Engineering disciplines. The projects, technical advice, and financial support are provided by the corporate sponsors. Teams and individuals are evaluated against defined project deliverables and lecture/workshop performance. Since 1994, over 1400 students from 12 disciplines have participated in 242 projects from 62 sponsors, which include companies such as NASA, Boeing, Dell Computer, Dow Chemical, DuPont, Energizer, Florida Power Corp., General Dynamics, Harris, Honeywell, IBM, Kimberley-Clark, Kraft Foods, Lockheed Martin, Motorola, Pratt & Whitney, Raytheon, Siemens, Southern Nuclear, Sunbeam, Texas Instruments, Tropicana, and the US Air Force.

7. CONCLUSIONS

We have described a set of tools for intelligently managing infrastrucutre construction projects, developed by a group of undergraduate students at the University of Florida, under the IPPD (Integrated Product and Process Design) program, and sponsored by InfoTech, Inc. We have describe the two-semester design and implementation effort and the most significant results, including multi-state database access, identification of similar but distinct construction items, and a relevance clasification scheme that significantly reduces the number of items found. Our understanding is that InfoTecj, Inc. intends to develop these tolls further, and deploy them as part of their BidExpress software.

REFERENCES

[1] http://www.infotechfl.com/

[2] https://www.bidx.com/index.html

[3] http://www.ippd.ufl.edu

[4] http://www.ruby-lang.org/en/

[5] http://rubyonrails.org/

[6] http://rubygems.org/gems/activemodel

[7] http://haml-lang.com/docs/yardoc/file.HAML_REFERENCE.html

[8] http://www.mysql.com/

[9] http://nokogiri.org/

[10] http://rake.rubyforge.org/

[11] http://rubygems.org/gems/rsolr

[12] http://outoftime.github.com/sunspot/docs/

Authorization and Disclaimer