Do you believe in life after OO and Java?
Then try these alternative projects.

Functional languages like ML, Charity, and Haskell are closer to mathematical notation than are imperative or object oriented languages. Expressions denote mathematical entities rather than defining the transitions of an abstract machine.
Looking beyond programming, CafeOBJ, Maude, Specware, and CASL are specification systems each offering their own particular perspective on the development process.
There are also a set of alternative data models and database systems on offer including FDM, ORM, and constraint databases.
The projects below provide an excellent training ground for your software skills. Some of the descriptions are more detailed than others. If you are interested in any topics in these areas please let me know (Patrick.Browne@comp.dit.ie).

1) A category calculator in CAML.

Category Theory (CT), like Set Theory, is a branch of mathematics that can be useful to computer scientists. Without appropriate support software it can be quite difficult to learn and use. The aim of this project is to construct a calculator for category theory operations. These are analogous to union and intersection in set theory. The calculator will allow the user to input data and view the results. If you are feeling adventurous you might provide a graphical output.
All the code already exists and is fully explained in the book Computational Category Theory by Rydeheard and Burstall. The book and code are freely available on the Web. No knowledge of category theory is required. This is a pure software engineering project moving existing code to a more modern setting. The existing code consists of a set of distinct operations that need to be integrated and combined with a menu system for input and possibly both text and graphics for output.
Research outcomes

1) A knowledge of a functional programming language (CAML).
2) A knowledge functional systems design, documentation and implementation.
3) Designing the input and output aspects of the system.
4) Understanding the ML module system.
5) An insight to category theory itself

2) A triangle drawing programming in CAML.

This project based on the work of Blelloch et al. The basic idea is to allow the user to construct 2-dimensional objects from simple triangles. Also the user should be able to query objects asking 'what is the boundary of object A?' or 'how many triangles in object B?'. The basic ML code is available, the GUI needs to be built. If you are really ambitious you might look at how the triangles could be stored and queried in a database, however this is not part of the basic project.
Reference: Persistent Triangulations by Guy Blelloch, Hal Burchy, Karl Crary, Robert Harper, Gary Miller, and Noel Walkington
Research outcomes

6) A knowledge of a functional programming language (CAML).
7) A knowledge functional systems design, documentation and implementation.
8) Designing the input and output aspects of the system.
9) Understanding the ML module system.
10) An insight into graphics programming

3) Charitable data structures
Another ‘life after Java’ project. The project involves the implementation of standard data structures such as queues, trees, stacks in language called Charity. Many of these data structures are provided.
See: http://pll.cpsc.ucalgary.ca/charity1/www/home.html
http://en.wikipedia.org/wiki/Charity_programming_language

Research outcomes
Charity offers a radically different way of computing. What makes Charity different? How do data structures constructed in Charity differ from those implemented in Java? What criteria could you develop to compare Charity to Java?

4) A Specware approach to software development.
Specware is a tool for building and manipulating a collection of related specifications. Specware is a design tool, a logic, a programming language, and a database. Specifications are the primary units of information. A specification describes a concept to some degree of detail. Specware is a general-purpose tool that you can use to develop specifications for any system or realm of knowledge. You can do this as an abstract process, with no reference to computer programming; or you can produce a computer program that is provably a correct implementation of a specification; or you can use the process to redesign an existing program. You can use Specware to:
• **Develop domain theories**
You can use Specware to do “ontological engineering” or “system design”
• **Develop code from specifications**
You can use Specware to develop computer programs from specifications.
• **Develop specifications from code**
You can use Specware for reverse engineering -- derive a specification from existing code.
This project seeks to establish how successful Specware is in the above tasks.
Research outcomes:
1) The ability to use and understanding the main features of Specware.
2) The development of one ore more small examples illustrating Specware features
3) The ability to select criteria for evaluating the various roles of specware.
4) A comparison with other approaches e.g. Formal Concept Analysis (FCA) for domain modeling or UML for program design.

5) Construction programs as algebras.
Another ‘life after Java’ project. It involves the implementation of a set of standard data structures such as queues, trees, stacks in Haskell. The project is based on a particular approach to design and programming which was suggested by Bird and deMoor in their book Algebra of Programming. They have developed a technique of program design. The aim of this project is to explore their technique and compare it to a Java/UML approach. All the required code for the data structures is provided. What makes algebraic program construction different?
Research outcomes
1) Be able to construct programs using the Bird and deMoor technique.
2) Develop criteria to compare Bird and deMoor’s approach to the UML/Java approach.

3) Be able to describe how data structures constructed in Haskell differ from those developed using UML/Java.
4) Open research questions. What diagrams and tools could be used?

6) Viewing Lifestyles or "watching the time go by".

This project is based on the work of Damir Medak. He has written some Haskell code that describes how geo-objects vary over time. His essential idea is that geographical objects such as buildings have a birth, life and death called 'lifestyle'. The objective of the project is to develop a graphical front end to Menak's basic algorithm. This GUI would simulate the passage of time displaying a small geographic scene (or map) as it varies over time. The original code is fairly small consisting of a few hundred lines. Nonetheless considerable software engineering skills would be required to organise the code into a coherent graphic based Haskell application. Initially simple graphics such as rectangles should be implemented. This project would also give the student the opportunity to incorporate map features such as roads, building, and counties.

Research outcomes

1) The ability to code a substantial application in Haskell.
2) The ability to re-engineer and re-use existing code to generate a new application.
3) The ability to understand and use the purely functional approach to programming.

Reference: Lifestyles - A Paradigm for the Description of Spatiotemporal Databases: by Damir Medak, 1999, Technical University Vienna

7) A formal approach to the analysis and design of E-voting systems using CafeOBJ.

This project has been taken up and completed by a DT249 student. However, there is room for <u>further development</u>. Systems that implement electronic voting require both technical analysis and public debate. This project is concerned with analysing technical qualities of e-voting systems. As ICT professionals, we should ensure that best practice is used in the analysis, design, and implementation of computer-based voting systems. This project requires:

1) Gathering of a set of requirements,
2) Distilling these requirements into a set of CafeOBJ specifications,
3) Refining the specification to design level,
4) Testing and proving certain properties of a subset of the design.

Learning outcomes

1) A deep understanding of the CafeOBJ specification language.
2) The ability to use all the specification and proving tools of CafeOBJ.
3) The ability to analyse a system from more that one perspective (e.g. structural and behavioural).
4) A good understanding of the technical issues involved in e-voting systems.
5) The ability to evaluate the practical value of formal techniques for systems analysis.

This project is not a social essay on how E-voting can enhancement or retard the democratisation of society.

8) Comparing the Geodirectory to Post Codes using PostgreSQL/PostGIS (or ORACLE).

This project involves a technical assessment of two alternative ways of linking address databases with geographical information systems (GIS). The project involves the use of PostgreSQL/PostGIS which allows the storage of both address information and spatial data. The project builds on the work of a K268 student, Isidor Pavia, who has loaded the sample version of the Geodirectory into Oracle and implemented some spatial queries.

The purpose of the project is to compare the existing Geodirectory with the newly proposed post codes. This is essentially a database project.

Some background

Geocoding is the process of relating a place names (or textual addresses) to a geographically referenced coordinate system. It is important for linking databases containing address to GIS. Geocoding enables government, local authorities, and business to know where the clients are. It is a fundamental process for geodemographics and will also be a significant part of any form of e-government strategy. Recently, a post code system has been proposed for Ireland. This is interesting because Ireland already has a Geodirectory which was set up by An Post and Ordnance Survey of Ireland. The Geodirectory provides a standardised, accurate database containing the precise geographic location (X, Y co-ordinates) and the postal address for every home and business in Ireland. Databases and mailing lists can be built with GeoDirectory references.

Research outcomes

1) The ability to build two prototypes, one for the Geodirectory and one for PostCodes.
2) A deep understanding of complex SQL particular spatial and data mining queries.
3) A deep understanding of both the Geodirectory and Post Code systems.
4) The ability to document complex systems.
5) The ability to develop applications the use geo-coding.
6) Research skills to develop a set of comparison criteria.

Reference

9) Simulating the Geographic Information with OCL and ASL.

The Open Geospatial Consortium (OGC) maintain the standards necessary for representing geographic data. The OGC standard is expressed using the UML (and some OCL). This project seeks to augment the OCG specifications with OCL and ALS. In particular the student should investigate how the two software tools USE 2.2, from Bremen University, and iUML from Kennedy Carter could be used to help clarify or formalise the OGC standards. This project is an investigation into how standards are specified.

Research outcomes

1) The ability to use the software tools i.e. USE and iUML.
2) An understanding of a large OO standard.
3) Be able to use the semi-formal approach to specification.
4) The ability to select from the OGC specification those parts which will benefit from a formal treatment and those section which are best left to human interpretation.

10) Set up and run a spatial-temporal database called TRIPOD
The main aim of this project is to install our national Irish Geographic database (scale 1:250,000) into TRIPOD. This is a spatial-temporal database capable of addressing spatial and temporal queries
Research outcomes:

1) To be able to execute spatial and temporal queries on the School of Computing Irish data set.
2) To explore the possibility of a Web front end for a graphical representation of the map data.

11) Digital photogrammetry.
The aim of this project is to be able to construct digital maps from aerial photographs and store the results in s spatial database. We have the main reference and software. Reference: Digital Photogrammetry A practical course by Linder, Wilfried, With a pair of 3-D glasses, ISBN: 3-540-00810-1

12)Map editor using Open Map.
A K268 student Declan Lynch wrote a database link from OpenMap to Postgis, which is a spatial extension to PostgreSQL. His main focus was establishing the JDBC link and developing spatial queries. This project aims to extend Declan's work to include map editing and updating.
Research outcomes.

1) The ability to set up a complex Java to PostgreSQL connection (this is not vanilla JDBC).
2) Be able to program and use a large open source Java application.
3) Working with existing systems develops the ability to address many practical software challenges.

13) Where is our school? (diploma project)
This project builds on a K268-3 project by Paul Duff. This is a web based GIS using Java, Geoserver, Postgis and PostgreSQL. The idea is to allow primary schools to enter their location and associated information into a web viewable map. Further the map itself could be queried allowing school children to enter school or course related information.

13) Where is our school? (ordinary degree project)
This project builds on a K268-3 project by Paul Duff. This is a web based GIS using Java, Geoserver, Postgis and PostgreSQL. The idea is to allow primary schools to enter their location and associated information into a web viewable map. Further the map itself could be queried allowing school children to enter school or course related information.

14) Bird and Tree Survey. (ordinary degree project)
The government agency Enfo require information on various natural recourses. The basic idea here is to enable school children or their teachers to enter the results of environmental surveys directly on to a web based maps and database. Technically similar to the previous project.

15) How can software engineers use proof systems.
This project essentially focuses on the Isabelle/HOL system and looks at how the software engineer can use such a system. It would involve setting up the software and configuring it to connect to CASL (a specification system). Research outcomes:
1) At analysis level. Existing CASL specifications can be tested or proved.
2) At design level. The ability to prove certain CASL architecture features are valid.

16) Temporal Interval Logic
Some facts appear to be eternally true (e.g.1+1=2). But many facts are constantly changing and are only valid for specific intervals of time (e.g. the sate of your bank balance). This project investigates temporal intervals as an important data structure. Further temporal intervals have given rise to their own logics (or reasoning system). This project requires the conversion of a specification written in PVS to the Isabelle/HOL system. We could include an investigation into existing interval software.

17) PostgreSQL/PostGIS for network querying.
This builds on the work of two students who developed a GIS using PostgreSQL/PostGIS. These students used PostgreSQL/PostGIS to query geographic data. Some features were identified in the structure of the data that limited the ability to perform network querying. This projects seeks to
1) Identify the issues in network querying.
2) Enable our current map data to be queried.
3) To further explore the network facilities of PostgreSQL/PostGIS.
See chapter 6 of
Shashi Shekhar, Sanjay Chawla (2003). Spatial Databases A Tour. Prentice Hall.
"Introduction to Spatial Databases: Applications to GIS" by Philip Rigaux, Michell Scholl and Agnes Voisard. Morgan Kaufmann 2001. on line at:

19) Formal Concept Analysis(FCA): Is a mathematical technique for conceptual data analysis and knowledge processing. It formalizes the terms context, concept and concept hierarchy. FCA is based on lattice theory and partial order. Lattices can be expressed in categorical terms. The roles for FCA include:
1) Integrating ontologies.
2) Analysis of the internal consistency of ontologies.

There is plenty of good software for this approach see Concept Explorer, see http://sourceforge.net/projects/conexp

20) CommUnity a language for program design.
This project is based on the work Jose Fiadeiro. The project explores the COMMUNITY Work Bench (CWB) and it associate methodology. The software can be downloaded from the COMMUNITY Workbench Web page http://ctp.di.fct.unl.pt/~co

The student should focus on how CWB handles specification, refinement, architecture, components and component interconnections.
References:
The main reference is the CWB web site http://ctp.di.fct.unl.pt/~co

21) Exploring some modern design approaches.
This project is based on the work of Lano and Fiadeiro. It explores two tools and their associated methodologies. These are CDE and RSDS. These can to be explored through the book called "Software Design Using Java 2" by Lano, Fiadeiro and Andrade, Published by Palgrave, ISBN 1-4039-0230-5. The software comes with the book and can generate Java code. The project would focus and compare on these approaches. CDE is based on the idea of a "coordination contracts" (not to be confused with *design by contract*). It would be interesting to compare these approaches with MDA ideas. A small case study will be provided. The main reference is:
"Software Design Using Java 2" by Lano, Fiadeiro and Andrade, Published by Palgrave, ISBN 1-4039-0230-5.

22) The Functional Data Model
The functional data model (FDM) can be seen as an alternative to the relational approach. This project explores the FDM and investigates how it handles queries and updates of complex date. Initially the student will study how simple geometric shapes can be stored and queried. The database *pfdm_lite* can be downloaded from:
http://www.csd.abdn.ac.uk/~pgray/FDMDownload.html
http://www.csd.abdn.ac.uk/~pfdm/user_manual/user_manual.html

Reference (supplied): An Analysis of Geometric Modeling in Database Systems
ALFONS KEMPER and MECHTILD WALLRATH
Universitat Karlsruhe, Institut fiir Znformatik ZZ, D-7500 Karlsruhe,

23) Modelling time and space in a constraint database.
The basic idea of a constraint data model is to generalize the relational model to include constraint formulae as a basic data type.
Here is an SQL query:
`select x + y from R where x < y;`
can be written as a constraint as follows:
`{x + y | R(x, y) and x < y}`
Even the basic tuple `T(x,y)` can be viewed as a constraint:
$(x = a_i)$ and $(y = b_i)$
This project will use the MLPQ/PReSTO constraint database systems to construct spatial and temporal queries. All the data and documentation is available. The executable file of the MLPQ/PReSTO system and sample files are available at: http://www.cse.unl.edu/ revesz.
This site also contains the reference material. The research could be extended to explore how the constraint model fits in with the current object oriented approaches such as the Open Geospatial Consortium (OGC).

24) CafeOBJ graphics.

This project involves producing drawings representing CafeOBJ source code. You should use some existing drawing program such as uDraw(Graph) uDrawGraph@informatik.uni-bremen.de

It may also be possible to use CafePie (see http://citeseer.ist.psu.edu/595976.html) Here is the general idea. Given the program (actually a specification) below:

```
mod* ACCOUNT {
protecting(INT)
*[ Account ]*
op init : -> Account
bop balance : Account -> Nat
bop _deposit_ : Account Nat -> Accountbop _withdraw_ : Account Nat -> Account
var N : Nat
var A : Account
eq balance(init) = 0 .
eq balance(A deposit N) = balance(A) + N .
cq balance(A withdraw N) = balance(A) - N if N <=  balance(A) .
cq balance(A withdraw N) = balance(A) if balance(A) < N . }
```

We get the following diagram: