Project Suggestions

Here are a number of project suggestions grouped under 6 general categories for your convenience. Each of these suggestions is fairly general. It is up to your group to precisely define and scope your project to reflect your interests. *Remember, these are just suggestions and you are free to modify them in any way you think is interesting.*

The categories are:

1. *Software components*
   These are either software components which can be used directly as part of other systems or generators which can generate program code or data.

2. *Software engineering tools*
   These are tools to support specific software engineering activities such as requirements, design, etc.

3. *University support systems*
   These are software systems to support some functions of a University department.

4. *Information management systems*

5. *Simulation systems*
   These are systems which simulate some other machine.

6. *Web systems*

**Software components**

**A set of components to support graphical editing**
Graphical editing systems involving the manipulation of simple graphical objects such as circles, rectangles, etc. are part of many tools and a basic component set may be used to simplify the construction of these editors. The aim of this project is to build a set of objects or abstract data types which may be used as a basis for constructing graphical editors. You should include components to represent rectangles, ellipses, polygons with the ability to connect these using lines of different thicknesses. It should be possible to colour entities, move them around, re-size them, group and ungroup them, etc.

**An integrated help and message system**
This system should be designed as a set of abstract data types so that it may be incorporated as a component in other systems. Your design should take into account that the system may have to present help and messages at different levels of detail and in different languages.

**A screen and forms generator.**
The objective of this project is to build a system which allows the user to create form and screen designs interactively by specifying fields on the form, their contents and their
Software tools

Program analysis tools
The objective of this project is to build static and/or dynamic program analysis tools. These may provide static analysis facilities as discussed in Chapter 24 of your text or dynamic analysis showing how often each statement in the program has been executed. You can add further analysis facilities such as a facility which shows where names defined in the program are used and a facility which analyses the dynamic storage allocation in the program (all store which is allocated should be de-allocated). An abstract syntax tree is necessary for this project.

Program visualization tools
The objective of this system is to generate visual representations of a program in some programming language. There are various representations that could be developed such as a graphical representation of the call structure of the program, a representation of the composition structure, a representation of the scope hierarchy, etc. For object-oriented languages, you could generate service-usage charts or aggregation charts. This type of system has tremendous scope from simple, single-window systems to complex multi-window systems with a range of display options. An abstract syntax tree is necessary for this project.

A compiler testing system
The objective of this project is to build a compiler input generator which may be used to test compilation systems. You use the defined language grammar to generate strings representing correct and incorrect programs and integrate this with a system which submits input to a compiler and processes the compiler output to check that syntactically-correct strings have been parsed without error messages and that error messages have been generated for incorrect programs.

Design workbenches
There is a range of possible projects where design workbenches of various types could be built. These can include the components discussed in Chapter 26 of your text such as design editors for different representations, report generators, code generators, etc.

CBSE support system
The aim of this project is to build a software components catalogue which users may browse to discover reusable software components. It may be accessed using keywords associated with each component. It must be possible to enter and classify components. The interface to the system may either be a query language interface, or a forms-based interface. An extension to the project is to implement a web-based graphical browsing capability for the components catalogue and to allow components to be downloaded from remote sites.
System change control and reporting system
This system is intended to support a change reporting and control system as discussed in Chapter 32 of your text. It should allow changes to be logged, submitted for approval and subsequently tracked. It might be integrated with an electronic mail system so that the change reports can be passed automatically to the change control board.

A design rationale system
Design rationale is concerned with providing reasons why particular design decisions are made. A design rationale system allows proposals to be made and arguments for and against these proposals to be maintained. You might also include the notion of risk in such a system where as well as simply putting forward a design proposal, the proposer should also be able to describe risks and associate some numeric value with these risks. The rationale system should be able to assist with decision making based on these risk factors as well as technical arguments.

University support systems

A student record system
The aim of this project is to maintain a student record system maintaining student records within a single University or College department. The system should allow personal details to be recorded as well as classes taken, grades, etc. It should provide summary facilities allowing information about groups of students to be retrieved. Assume the system is intended for use by departmental administrative staff with no computing background.

A laboratory management system
This system is intended to support the administration of an undergraduate laboratory where different equipment is available to students for different experiments at different times. The system should assist with experiment scheduling and keep records of equipment purchases, reliability and maintenance.

A coursework submission system
This system should manage the electronic submission of assignments from students to instructors. It should keep track of which students have submitted assignments and should allow instructors to manage the grading of assignments. It should provide summaries of student grades and facilities for electronically returning information to students. You might consider some kind of annotation system where instructors could comment on electronic versions of students’ work. You could also, for example, include encryption facilities in the system.

Information management systems

Group diary and appointments system
This project is concerned with providing a shared diary where appointments for a group are recorded. When a meeting involving more than one group member is required, their
diaries are consulted and a mutually suitable time discovered. Appointments may be moved to make a suitable time. Note that some appointments are not movable.

**Ideas processor**
This project is concerned with providing a means of organizing ideas. It should support the input of headings, the moving of headings, the association of text with headings, etc. Do not assume that a simple hierarchical structure is appropriate. It should be possible to form a network of ideas. Make use of graphical capabilities if these are available.

**Personal data management system**
The aim of this project is to build a small database management system used to record personal information such as details of records and books, information about recipes and so on. Their may be a diversity of different kinds of information recorded and the system should be usable by those without a computing background. One possible approach would be to build a Macintosh-like HyperCard system.

**Photograph library system**
This system should record and retrieve information about photographs in a library. It should support keyword classification and retrieval requests for different types of photograph. Again, the system should be designed for use by people who are not familiar with computing concepts. This should be developed on a personal computer system and you could use some public-domain Photo-CD systems to test the system. Depending on the implementation platform, you may wish to include various image manipulation facilities in the system.

**Simulation systems**

**Memory management simulation**
A virtual memory system may be based on fixed-size pages which are moved between RAM and disk. The objective of this project is to build a memory management simulator to test the effects of different page replacement strategies, page sizes, etc. The system should have facilities to display unused memory and to illustrate the page traffic to and from disk. It should be driven by simple commands which request and release pages in a particular system. The system may have a simple textual or a graphical interface associated with it.

**Drug delivery system**
This project simulates a drug delivery system as discussed in Chapter 21 of your text. The different parts of the system should be simulated by separate processes and it should be possible to load different controlling software into each system component to test its operation.

**Web systems**

**Web site checking and repair**
Many web sites have broken links and the associated filestore contains unused and unwanted files. The aim of this project is to develop a system that checks a web site for errors of this type and which provides a simple control interface for users to repair these errors. The system should 'learn' as the user repairs an error so that if a link is replaced by another link on one occasion, the system should automatically propose this change if a further error involving the same files is detected.

**Web site cross-referencing**

The aim of this project is to develop a cross-referencing system that can traverse a web site and produce a table showing the links that are used on each web page and to show every web page on a site that references a particular web page. Limited external cross referencing is also possible where you may follow an outgoing link and, if this has a back link, you can then cross-reference that site.

**Web page layout of Java programs**

In this project, your aim is to take a Java program laid out as text and to convert it into a coherent set of web pages where the Java is displayed in HTML. You need to be able to find all of the objects referenced in the program (normally in separate files) and to display these in separate pages. You may create several different indexes into the Java program pages. Your system should deal with the layout of Java programs of up to 10,000 lines of code. To do this project, you may have to parse the Java code to discover the objects used. There are a number of free Java components for parsing available that can be downloaded.