

# **CADDIE Expert System**

---

**Making the right decision on the golf course just got easier!**

## Abstract

This paper describes the project life cycle model for the CADDIE Expert System. Included in this model are the project planning, knowledge definition, knowledge design, and implementation details.

1	Planning.....	1
1.1	Tasks.....	1
1.1.1	Feasibility Assessment.....	1
1.1.2	Resource Management.....	1
1.1.3	Task Phasing.....	1
1.1.4	Schedules.....	1
1.1.5	Preliminary Functional Layout.....	1
1.1.6	High-Level Requirements.....	1
2	Knowledge Definition .....	2
2.1	Source Identification and Selection.....	2
2.1.1	Tasks.....	2
2.2	Acquisition, Analysis, and Extraction.....	2
2.2.1	Tasks.....	2
3	Knowledge Design .....	4
3.1	Definition.....	4
3.1.1	Tasks.....	4
3.2	Detailed Design.....	4
3.2.1	Tasks.....	4
4	Code and Checkout .....	6
4.1	Tasks.....	6
4.1.1	Coding.....	6
4.1.2	Tests.....	6
4.1.3	Source Listings.....	6
4.1.4	User's Manual.....	6
4.1.5	Installation/Operations Guide.....	6
4.1.6	System Description Document.....	6
5	Knowledge Verification .....	7
5.1	Formal Test.....	7
5.1.1	Tasks.....	7
5.2	Test Analysis.....	7
5.2.1	Tasks.....	7
6	System Evaluation .....	8
6.1	Tasks.....	8
6.1.1	Results Evaluation.....	8
6.1.2	Recommendations.....	8
6.1.3	Validation.....	8
6.1.4	Interim or Final Report.....	8

...following figure 6.7 from Giarratano and Riley...

// TODO: Insert/Create Figure 6.7 from Giarratano and Riley

# **1 Planning**

## **1.1 Tasks**

### **1.1.1 Feasibility Assessment**

*Determine if it is worthwhile to build the system and, if so, whether expert system technology should be used.*

### **1.1.2 Resource Management**

*Assess resources of people, time money, software, and hardware required. How to acquire and manage the required resources.*

### **1.1.3 Task Phasing**

*Specify the tasks and their order in the stages.*

### **1.1.4 Schedules**

*Specify the starting and delivery dates of tasks in the stages.*

### **1.1.5 Preliminary Functional Layout**

*Define what the system should accomplish by specifying the high-level functions of the system. This task specifies the purpose of the system.*

### **1.1.6 High-Level Requirements**

*Describe in high-level terms how the functions of the system will be accomplished.*

## **2 Knowledge Definition**

### **2.1 Source Identification and Selection**

#### **2.1.1 Tasks**

##### **2.1.1.1 Source Identification**

*Who and what are the knowledge sources, without regard to availability.*

##### **2.1.1.2 Source Importance**

*Prioritized list of knowledge sources in order of importance to development.*

##### **2.1.1.3 Source Availability**

*List of knowledge sources ranked in order of availability. Books and other documents are generally much more available than the human experts.*

##### **2.1.1.4 Source Selection**

*Select the knowledge sources based on importance and availability.*

### **2.2 Acquisition, Analysis, and Extraction**

#### **2.2.1 Tasks**

##### **2.2.1.1 Acquisition Strategy**

*Specify how knowledge will be acquired by methods for interviewing experts, reading documents, rule induction, repertory grids, and so forth.*

##### **2.2.1.2 Knowledge Element Identification**

*Pick out the specific knowledge from sources that will be useful in this iteration of the life cycle.*

##### **2.2.1.3 Knowledge Classification System**

*Classify and organize the knowledge to aid in knowledge verification and understanding by developers. Use hierarchy groups whenever possible.*

### **2.2.1.4 Detailed Functional Layout**

*Specify the functional capabilities of the system in detail. This is at a more technical level whereas the preliminary functional layout was at a managerial level.*

### **2.2.1.5 Preliminary Control Flow**

*Describe general phases that the expert system will execute. Phases correspond to logical rules that are activated/deactivated in groups to control execution flow.*

### **2.2.1.6 Preliminary User's Manual**

*Describes system from user's viewpoint. An often ignored but essential part of the system. It is absolutely important to involve users as soon as possible for feedback. If they don't use the system, it's worthless.*

### **2.2.1.7 Requirements Specifications**

*Define exactly what the system is supposed to do. The expert system will be validated using these requirements.*

### **2.2.1.8 Knowledge Baseline**

*Baseline knowledge for the system. Any changes must now be done by a formal change request. The high-level knowledge is now adequate for the next stage of knowledge design.*

## **3 Knowledge Design**

### **3.1 Definition**

#### **3.1.1 Tasks**

##### **3.1.1.1 Knowledge Representation**

*Specify how knowledge will be represented, such as rules, frames, or logic. Dependent upon what the expert system tool will support.*

##### **3.1.1.2 Detailed Control Structure**

*Specify three general control structures: 1. If the system is embedded in procedural code, how it will be called. 2. Control of related groups of rules within an executing system. 3. Metalevel control structures for rules.*

##### **3.1.1.3 Internal Fact Structure**

*Specify the internal structure of facts in a consistent manner to aid in understanding and good style.*

##### **3.1.1.4 Preliminary User Interface**

*Specify a preliminary user interface. Get feedback from users about the interface.*

##### **3.1.1.5 Initial Test Plan**

*Specify how code will be tested. Define test data, test drivers, and how test results will be analyzed.*

### **3.2 Detailed Design**

#### **3.2.1 Tasks**

##### **3.2.1.1 Design Structure**

*Specify how knowledge is logically organized in the knowledge bases and what is in the knowledge base.*

### **3.2.1.2 Implementation Strategy**

*Specify how the system is to be implemented.*

### **3.2.1.3 Detailed User Interface**

*Specify the detailed user interface after receiving user feedback from the preliminary user interface design.*

### **3.2.1.4 Design Specifications and Report**

*Document the design.*

### **3.2.1.5 Detailed Test Plan**

*Specify exactly how the code will be tested and verified.*

## **4 Code and Checkout**

### **4.1 Tasks**

#### **4.1.1 Coding**

*Implement coding.*

#### **4.1.2 Tests**

*Test code using test data, test drivers, and test analysis procedures.*

#### **4.1.3 Source Listings**

*Produce commented, documented source code.*

#### **4.1.4 User's Manual**

*Produce working user's manual so experts and users can provide feedback on system.*

#### **4.1.5 Installation/Operations Guide**

*Document installation/operation of system for users.*

#### **4.1.6 System Description Document**

*Document overall expert system functionality, limitations, and problems.*

## **5 Knowledge Verification**

### **5.1 Formal Test**

#### **5.1.1 Tasks**

##### **5.1.1.1 Test Procedures**

*Implement formal test procedures.*

##### **5.1.1.2 Test Reports**

*Document test results.*

### **5.2 Test Analysis**

#### **5.2.1 Tasks**

##### **5.2.1.1 Results Evaluation**

*Analyze test results.*

##### **5.2.1.2 Recommendations**

*Document recommendations and conclusions of tests.*

## **6 System Evaluation**

### **6.1 Tasks**

#### **6.1.1 Results Evaluation**

*Summarizes the results of testing and verification.*

#### **6.1.2 Recommendations**

*Recommend any changes to the system.*

#### **6.1.3 Validation**

*Validate that the system is correct with respect to user needs and requirements.*

#### **6.1.4 Interim or Final Report**

*If the system is complete, issue final report. If not, issue an interim report.*