SIGCSE *Special Projects* Grants

2003 March

Application Form

Applicant Name: Richard E. Pattis Date: 3/26/2002

Institution: Carnegie Mellon University

Postal Address: Computer Science Department
5000 Forbes Avenue
Pittsburgh, PA 15213

Email Address: pattis@acm.org

SIGCSE (ACM) Membership Number: XXXXXXX

Project Title: A Framework for Playing Network Games in CS1/CS2

Brief Project Description (under 50 words):

I'll develop a Java framework (and JavaDoc API) allowing CS1/CS2 students to write games played over a network. Games are hosted on one machine (storing shared state). Manual or automated players on other machines command/query it via strings. I'll write several assignments and sample games to distribute with this framework.

Amount Requested in U.S. Dollars: $3200

Statement: Java's extensive networking API provides a base for enormously interesting programming assignments, but its complexity easily overwhelms beginners: CS1/CS2 students. I have prototyped a small framework that allows beginners to develop network games more easily, allowing them to focus on implementing the games themselves.

The term "game" is too specific. The framework supports any application in which one server (machine) stores state, and clients (other machines) query its state or command it to change its state according to rules. Communication occurs via method calls and strings (hiding sockets and threads).

For example, I have written a chat room server and client in this framework. The server allows clients to issue commands: to login, create chat topics, join chats, leave chats, and send/receive messages to/from all clients in a chat; the server allows clients to issue queries to find active chat topics and who has joined them.

To use this framework, students extend two classes: **Game** and **PlayerProxy**. These superclasses provide general game services (such as player registration); the subclasses written by the students provide game-specific command/query protocols and semantics. Finally, students write a remote **Player** class that accepts commands from a manual player (via the console or GUI) and displays the results of queries (ditto); a more sophisticated **Player** class would generate commands and interpret the results of queries automatically.

In practice, instructors can supply any combination of the **Game**, **PlayerProxy**, or **Player** subclasses, asking the students to write the other components of the application

Resources: I propose to use this grant to fund one CMU undergraduate (CS major) to work with me throughout June and July 2003. He/She will continue developing the framework (game formation, turn-taking games, speed enhancements, etc.), write all its JavaDoc, write a few well-commented sample games (some using a GUI), and prepare the materials for easy dissemination. The student will earn $10/hour, 40 hours/week, over 2 months (10*40*8 = $3,200). I will meet with the student daily and write two assignments/solutions using the framework.

Evaluation: Next month, I will assign my current CS1 students to implement the chat program discussed above using the prototype framework. During the fall semester, I will teach CS1 to 75 CS majors and 50 non-majors and Don Slater will teach 100 non-majors. We will assign one network game programming project in each class (I'll lobby other CMU lecturers to join us to provide feedback).

Dissemination: I will release these materials at the end of the summer and update the release after the winter break, based on fall semester feedback. The framework will include java source files, precompiled .jar files, and JavaDoc pages for the framework. It will also include sample applications (**Game**/**PlayerProxy**/**Player** classes) and the two programming assignments that I will use for evaluation in the fall. All materials will be available on my web site (announced via SIGCSE and AP-CS mailing lists), and I will submit them to educational repositories. I will be happy to discuss these materials at SIGCSE (CMU will pay for me to go).