

Request for Proposal – Client/Server Architecture

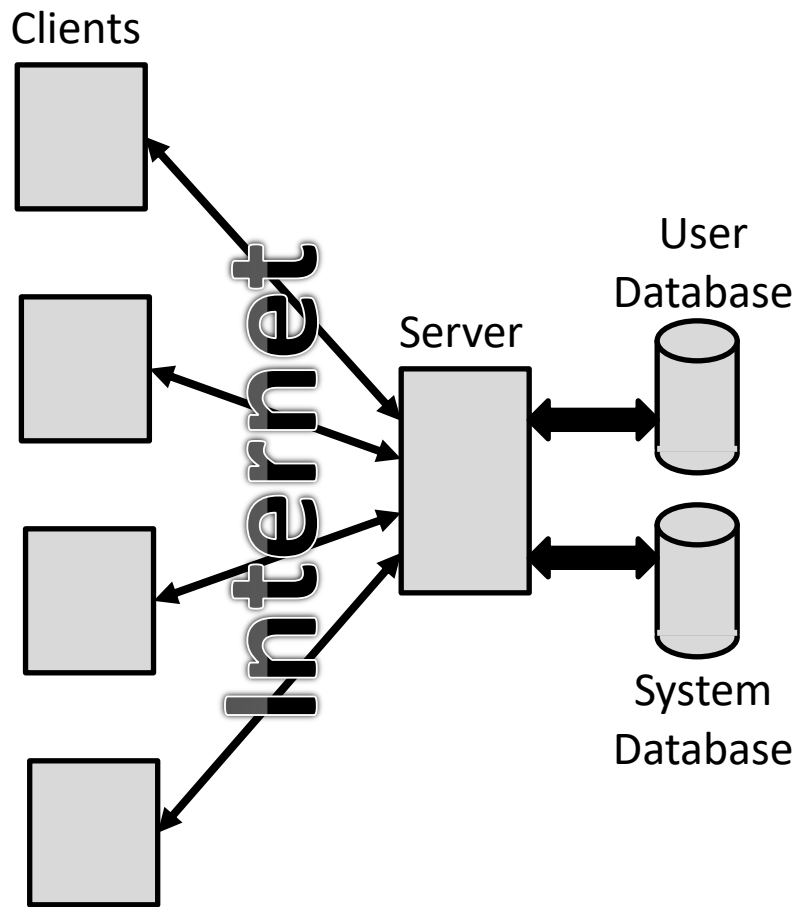
General Description

I am looking to develop a generic client/server architecture that includes the ability to store information in a database and generate email. The actual application to which this will be applied is not specified at this time.

System Type

The client is a standalone graphical user interface application to be run on a desktop or laptop computer. User input will be provided utilizing a keyboard and mouse. Server will have an administrative graphical user interface. The server will also run on a desktop or laptop computer (different from the client) and will have a graphical user interface for system monitoring. The server must be able serve multiple clients at any given time.

System Architecture



Specifications

- User database will hold (at a minimum) login authentication information
 - Username
 - Password
 - Email address
- System database will hold all required application persistent data
 - To be defined when the business logic is specified
 - Can store any data you choose at this time so as to demonstrate operability
- Graphical user interface (on the client) will allow the following functionality
 - New user registration
 - Username
 - Password
 - Enforce minimum length
 - Enforce combination of letters/numbers
 - Email address
 - Enforce proper form e.g. username@domainname.typ

- Login
 - Username
 - Password (protected)
 - Authenticate
 - Limit number of attempts to 3
 - After 3 attempts user becomes locked out and must request a new password be emailed to the email address on record (see password recovery below)
- Logout
- Password recovery
 - Username
 - Send password to email address on record
- Connect
 - User supplies the server IP address
- Disconnect
 - Disconnects the client from the server but leaves the client running
- Change password
 - User is logged in
 - User supplies a new password (specification above)
 - User supplies a retype of the password with must match the first for the change to take place
- Business Logic
 - To be specified at a later date
- Graphical user interface (on the server)
 - Query number of registered users
 - Query number of logged in users
 - Query which users are logged in
 - Query which users are locked out
 - Query number of connected users

Proposed Technologies

- Programming language
 - Java
- Database
 - MySQL
 - Mongo-DB
 - Flat file
- Email
 - Gmail SMTP protocol
- Client/Server Networking
 - Single-threaded client

- Multi-threaded server
- Graphical User Interfaces
 - JavaFX or Java Swing

Deliverables

Deliverable documents should always be of “professional” quality. They should: 1) be pleasing to the eye; 2) be free of grammatical errors; and 3) include all requested information. These documents will define your “identity” as viewed by your customer. You may choose the format that best suits your style as long as all requested information is included. Documents should be submitted to Blackboard. Document file format must be PDF.

First deliverable contents must include:

- Introduction
 - A brief description of the system to be designed and implemented
 - This should [essentially] restate the original problem as presented by the customer in a brief, abbreviated form
 - Use your own words
 - Don’t include all the details, just the high level (primary) items
- System perspectives for the point of view of:
 - The customer
 - The user
 - The development team

Note that the customer may not be the user. The customer specifies the system based on some product they have in mind while the user is the person who will be sitting down in front of the system. The developer is, of course, your team.

- Development environment to be used
 - What programming language will you use?
 - E.g. Java, Python, Visual Basic, C++
 - What development environment will you use?
 - E.g. Eclipse, IntelliJ, PyCharm, etc.
 - What GUI library will you use?
 - E.g. JavaFX, Java Swing, Tkinter, etc.
 - What email server will you use?
 - E.g. Outlook, Gmail, Yahoo, etc.
 - What database management system will you use?
 - E.g. Mongo-DB, MySQL, flat file, etc.

Provide justification/reasoning behind our choices.

- List of concerns regarding the specification

- E.g. Is there something about this proposal that makes you nervous?
- List of questions/clarifications regarding specification
 - Items in the proposal that you do not understand or for which you would like clarification
- Biography for all team members individually
 - Name
 - Programming experience
 - General statement of programming experience
 - Description of the largest (most complex) program that you have written