

Go In Action 1

Assignment

All rights reserved. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of the author.

Trademarked names may appear in this document. Rather than use a trademark symbol with every occurrence of a trademarked name, the names are used only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The information in this document is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this document, the author shall not have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this document

1. Objective

This assignment is to test your understanding and ability to apply the key concepts learnt in Go In Action 1, namely Networking, Templates and Dependencies.

2. Introduction

You are required to develop convert an existing simple prototype of an application in Go programming language, preferably the prototype submitted in Go Advanced, to a Client-Server setup.

3. Basic Requirements

Your application must:

- Incorporate the use of Server and Client using a localhost at port 5221.
- Use different **SIMPLE** selection pages of the application in `.gohtml` format.
- Preserve or improve upon existing prototype submitted if used, to adopt to the inclusion of a HTTP server.

Suggested application examples:

Example 1: Venue Booking System

Client	Server
Login Feature <ul style="list-style-type: none"> • New Customer sign up • Existing returning Customer • Admin 	Login <ul style="list-style-type: none"> • Issue Cookie • Store Session – User Information • Store User information
Customer Page <ul style="list-style-type: none"> • Browse venues • Search for available venues • Book venue • Edit booking • Edit Customer Info 	Venue Data <ul style="list-style-type: none"> • Venue availability • Booked venue details
Admin Page <ul style="list-style-type: none"> • Edit booking • Delete Sessions • Delete Users 	

Example 2: Dental Appointment system

Client	Server
Login Feature <ul style="list-style-type: none"> • New Patient sign up • Existing returning Patient • Admin 	Login <ul style="list-style-type: none"> • Issue Cookie • Store Session – User Information • Store User information
Customer Page <ul style="list-style-type: none"> • Search for available appointments • Book appointments • Edit appointment details • Edit Customer Info 	Venue Data <ul style="list-style-type: none"> • Appointment availability • Booked appointment details
Admin Page <ul style="list-style-type: none"> • Edit appointment details • Delete Sessions • Delete Users 	

4. Additional Features (bonus)

In addition to the basic requirements outlined, the complexity, comprehensiveness and user friendliness of the application will determine how well you score in this section.

Examples of issues to consider:

- User-friendliness, ease of use of Interface
- How appropriate were client-server implemented?
- How comprehensive is your program's error handling and handling of panics?
- How well your program supports concurrency?
- Good programming style, documentation.

6. Deliverables

Assignment Submission

You are required to upload file called **FullName_GoInAction1.zip** through the [Assignment Submission](#) link in GoSchool.

The zip file should contain

- All the necessary **files needed to run your application/game**.
- A short write-up in Microsoft word document format that clearly indicates info on:
 - Description of the client-server in general.
 - Description of at least **3** important main feature related to concepts **discussed in Go In Action 1.**
 - Description of at least **3** error handling and concurrency mechanism incorporated.