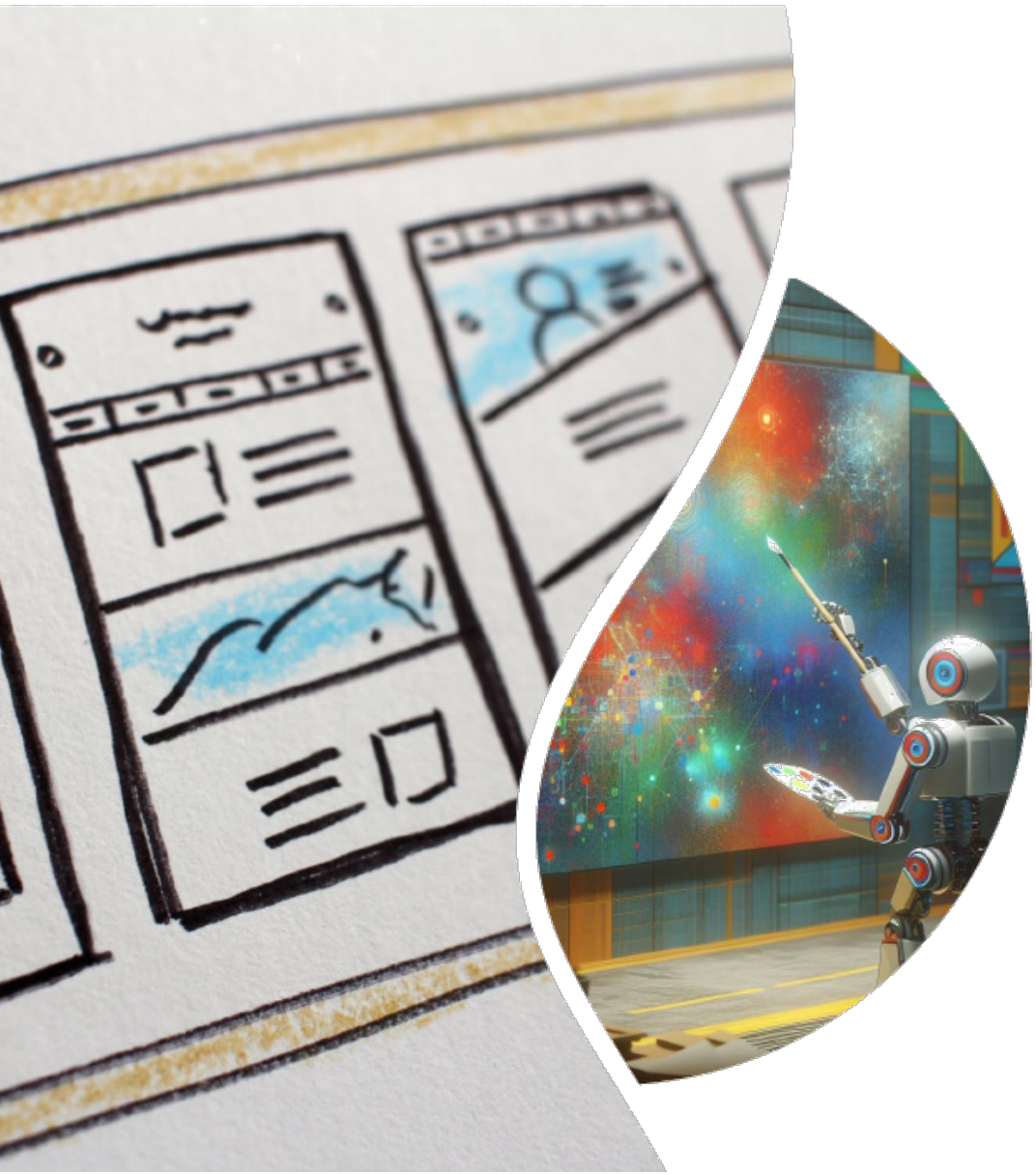


DGIN 5201
Digital Transformation
Lecture 2

Building a Front End

Vlado Keselj and
FE. Bordeleau

Time and date:
13:05–14:25, 9-Jan-2025
Location: LSC C236

Image: DALL-E. Bing Image Creator. Generated by AI

Previous Lecture

- Course Information
- Digital Transformation — One View
- Why two instructors
- Learning outcomes
- Delivery, topics, evaluation, policies
- Overview of the Business part
- Overview of the Computer Science part

Technical Foundations

- Two main foundations of Digital Innovation:
  - Computer as a general computing and information processing device
  - Internet as a general communication infrastructure
- Computer as foundation
  - File system, processes, users
  - Operating system, shell (bash)
  - Programs, utilities, commands, applications
- Internet and communication

The Evolution of the Internet: 1961–The present

- Early Innovation Phase, 1961–1974
  - Creation of fundamental building blocks
  - 1973–74: TCP/IP
- Institutionalization Phase, 1975–1994
  - Large institutions provide funding and legitimization
  - 1986, beside ARPANET, NSFNET began (civilian Internet)
- Commercialization Phase, 1995–present
  - Private corporations take over, expand Internet backbone and local service

More Detailed History of Internet and Web

Early Innovation Phase

1961 — Leonard Kleinrock (MIT) publishes a paper on packet switching networks.

1971 — E-mail is invented by Ray Tomlinson (BBN). Larry Roberts writes the first e-mail utility program.

1973 — Bob Metcalfe (XeroxPark Labs) invents Ethernet and local area networks; client/server computing invented

1974 — "Open architecture" networking and TCP/IP concepts presented in a paper by Vint Cerf (Stanford) and Bob Kahn (BBN).

1980 — TCP/IP officially adopted by DoD; Personal computers invented

Institutionalization Phase

1984 — DNS (Domain Name System) was introduced.

1989 — Tim Berners-Lee (CERN, Switzerland) proposes World Wide Web (HTML and HTTP).

1990 — Internet becomes available to wider public, ARPANET transforms to NSFNET.

1993 — Mosaic, the first graphical Web browser implemented by Mark Andreesen and others (National Center for Supercomputing at the University of Illinois).

Commercialization Phase

1995 — Commercial Internet born: commercialization of the US backbone, Network Solutions takes over domain registration.

1995 — Amazon founded by Jeff Bezos; AuctionWeb (eBay) by Pierre Omidyar.

1998 — Google founded by Larry Page and Sergey Brin.

2004 — Facebook founded by Mark Zuckerberg, Eduardo Severin, Dustin Moskovitz, and Chris Hughes.

2009 — Internet-enabled smartphones become a major extension.

Summary

- This was an overview of the Technical (CS) Part
- We will now take another look at a course calendar overview
- and start the unit on Rapid Prototyping

Course Calendar Overview

2025	Mo	Tu	We	Th	Fr	Sa	Su	
Jan	.6	7	8	9	10	11	12	(w1) Intro
	13	14	15	16	17	18	19	(w2) Rapid Prototyping
	20	21	22	23	24	25	26	(w3)
Feb	27	28	29	30	31	1	2	(w4) Disruptive Innovation
	3	4	5	6	7	8	9	(w5)
	10	11	12	13	14	15	16	(w6)
	17	18	19	20	21	22	23	(study break)
Mar	24	25	26	27	28	1	2	(w7) Emerging topic 1
	3	4	5	6	7	8	9	(w8) Emerging topic 2
	10	11	12	13	14	15	16	(w9) Emerging topic 3
	17	18	19	20	21	22	23	(w10) Emerging topic 4
	24	25	26	27	28	29	30	(w11) Reserved
Apr	31	1	2	3	4	5	6	(w12) Final Presentations
	.7	8	9	10	11	12	13	Report and code

Deliverables and Project Calendar Overview

2025	Mo	Tu	We	Th	Fr	Sa	Su	
Jan	.6	7	8	9	10	11	12	(w1) Intro
	13	14	15	16	17	18	19	(w2) Rapid Prototyping
	20	21	22	23	24	25	26	(w3)
Feb	27	28	29	30	31	1	2	(w4) Disruptive Innovation
	3	4	5	6	7	8	9	(w5) Assignment 1 due
	10	11	12	13	14	15	16	(w6)
	17	18	19	20	21	22	23	(study break)
Mar	24	25	26	27	28	1	2	(w7) Assignment 2 due, prj.spec
	3	4	5	6	7	8	9	(w8) Seminar report 1
	10	11	12	13	14	15	16	(w9) Sem.report 2, early proto.
	17	18	19	20	21	22	23	(w10) Sem.report 3
	24	25	26	27	28	29	30	(w11) Sem.report 4, prj.demo
Apr	31	1	2	3	4	5	6	(w12) Final Presentations
	.7	8	9	10	11	12	13	Report and code

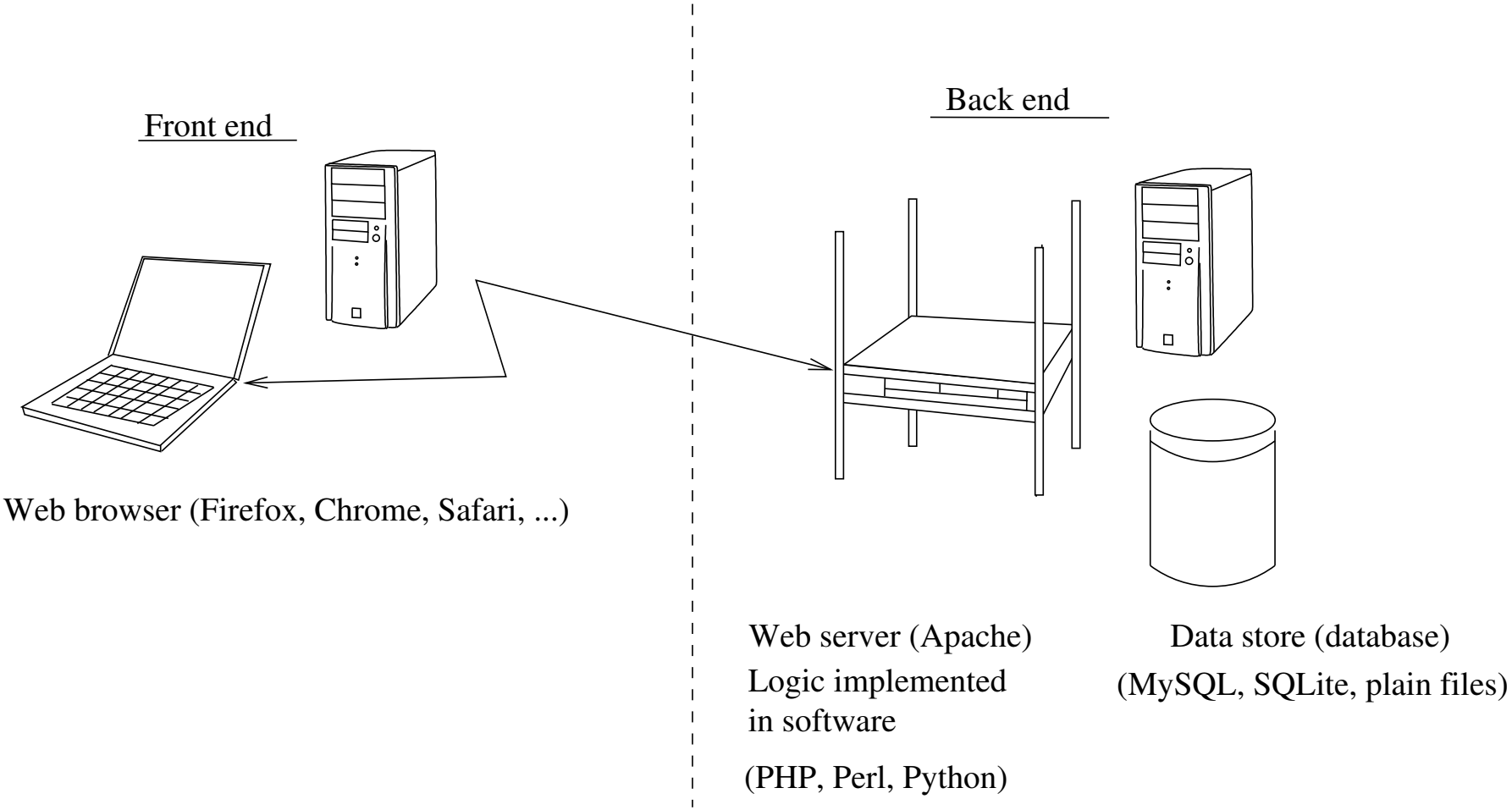
Unit Description

- Implementing a solution: Rapid prototyping
- Review of programming and Web fundamentals
- Hands-on exercises in fundamental technology
- Elements of building a three-tier system
- Techniques for rapid prototype building

Building MVP Example

- How to build an MVP (Minimal Viable Product) in a short time?
  - useful in the Rapid Prototyping model of development
- Course project requires a Three-Tier Architecture
- Three-Tier Architecture:
  1. User interface
  2. Control logic
  3. Data store

Three Tier Architecture



Three–Tier Architecture

Features of Three-Tier Architecture

- Front-end
  - HTML and CSS in a simple form
  - Improvements: JavaScript, AJAX; jQuery, and JS frameworks
- Back-end, logic tier
  - Scripting languages (PHP, Perl, Python, Ruby, etc.)
  - Straightforward: Apache and CGI
  - Improvements: Web frameworks such as Flask, Django, Mojolicious
- Back-end, data store tier
  - Straightforward: plain files, MySQL, SQLite
  - More: MongoDB, Redis, other database systems

Our Approach in this Unit

- Work on hands-on exercises
- Covering concepts and theory
- Exercises aimed at `timberlea` server
- Use your CSID and password
- Use of web site:
  https://web.cs.dal.ca/~*YourCSID*

Some Background Items

- Check your CSID and password, helpful site: `https://csid.cs.dal.ca/`
- Helpful if you have experience in `ssh` login to `timberlea.cs.dal.ca`
- Mac or Linux: `ssh` can be used from terminal
- Windows: PuTTY can be used
- PuTTY can be installed from `https://www.putty.org/`

Baseline Implementation

- Assume diverse background knowledge and levels
- Baseline Implementation:
  - login to `timberlea.cs.dal.ca` using CSID
  - work with a shell; e.g., bash, basic Unix commands
  - use of a plain-tex editor: emacs, vi, vscode, or similar
  - use of HTML, scripting languages, JavaScript, CSS
  - plain files for persistant data, database
- Make sure to be familiar with your CSID: `https://csid.cs.dal.ca/`
- Use ssh or PuTTY to login to `timberlea.cs.dal.ca`

Using timberlea Server

- ssh login into timberlea.cs.dal.ca
- Windows: you can use the program PuTTY
  - other options available; e.g., MobaXterm
- On Mac: open a Terminal and type:

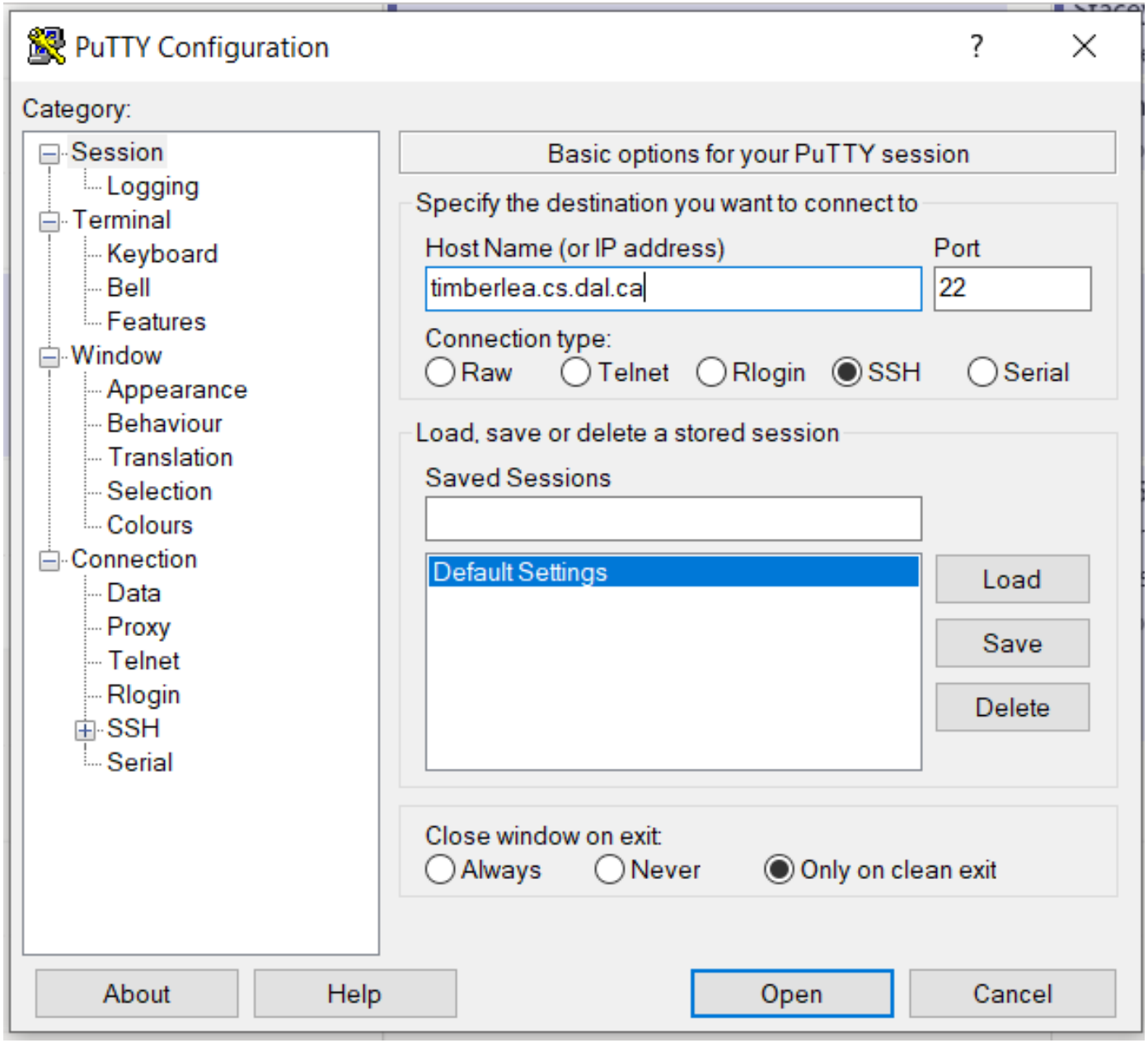
```
ssh <your_csid>@timberlea.cs.dal.ca
```

  where instead of <your_csid> you should use your own CSID
- On Linux: similarly to Mac, you open the terminal and type the same command:

```
ssh <your_csid>@timberlea.cs.dal.ca
```

Running PuTTY

- Double-click the PuTTY icon, and the following window should appear:


PuTTY Configuration
?
Category:
Session
Logging
Terminal
Keyboard
Bell
Features
Window
Appearance
Behaviour
Translation
Selection
Colours
Connection
Data
Proxy
Telnet
Rlogin
SSH
Serial
Basic options for your PuTTY session
Specify the destination you want to connect to
Host Name (or IP address)
Port
timberlea.cs.dal.ca
22
Connection type:
Raw
Telnet
Rlogin
SSH
Serial
Load, save or delete a stored session
Saved Sessions
Default Settings
Load
Save
Delete
Close window on exit:
Always
Never
Only on clean exit
About
Help
Open
Cancel

Hands-on Exercises

- You should use PuTTY or another client to login to `timberlea`
- FileZilla is a good tool to copy files back and forth, but does not provide access to command-line (shell)
- The following exercises should be finished and will be graded as a part of Assignment 1
- Example of command-line (bash shell) access:

Creating a Simple Web Page

- Try command: `pwd`
- Enter directory: `public_html`
- Create directories: `dgin5201/e1`
- Set permissions for this directory to be all-accessible: `chmod` command
- Go to directory `dgin5201/e1` and create file `index.html` with the following content:

```
<html><body>
This is a very small HTML file.
</body></html>
```

- Make `index.html` all-readable and access it over Web

Opening Web Page in a Browser

- Check that the page works; using Web browser open URL `https://web.cs.dal.ca/~<your_csid>/dgin5201/e1`
- Review the process of obtaining a web page

Concepts Review: Example 1

- `ssh` access, PuTTY, `bash` shell
- `bash` commands: `pwd`, `ls`, `cd`, `mkdir`, `chmod`, `rmdir`
- File permissions
- Text editors: `emacs`, `vi`, `pico`, `nano`, or use remote editing: FileZilla, `vscode`
- Emacs editor:
  `emacs index.html` or `emacs -nw index.html`
  `C-x C-s` to save, `C-x C-c` to exit, `C-h t` to go through simple tutorial (`C-` means Ctrl and other key)
- HTML: simple tags, `html`, `body`
- Web and HTTP access

Requirements of e1

- At the end of Example 1 (e1), there should be the following directories (folders), files and their permissions:

```
~/public_html/dgin5201                  rwx--x--x
~/public_html/dgin5201/e1               rwx--x--x
~/public_html/dgin5201/e1/index.html    rwxr--r--
```

- Content of `index.html` was given previously