

**School of Computer Science and Engineering
(CSE)**

**COMP9900 Information Technology Project
COMP3900 Computer Science Project**

2023 Term 3

Week 3

Dr Rachid Hamadi
r.hamadi@unsw.edu.au



UNSW
SYDNEY

Outline

- Interface Design Principles
- Git and GitHub
- GitHub Classroom
- GitHub and Jira Integration
- Project Proposal
- Week 3 Lab Tasks
- Q & A

Interface Design Principles

What is usability?

- When we say a software is **usable**, it means that it is **easy to use**
- Usability main elements
 - **Effectiveness**: Design fits its purpose
 - **Efficiency**: Amount of effort or time required to use
 - **Satisfaction**: Level of enjoyment of using the design



How to create an interface with a good usability?

For creating a user-friendly interface, one should:

1. Match between the system and the real world
2. Adapt the software to users' mental model
3. Consider consistency and standards
4. Prevent users' errors
5. Consider visibility of system components
6. Consider visibility of system status
7. Consider flexibility of system features
8. Help users recognize, diagnose, and recover from errors
9. Create help and documentation
10. Create an appropriate structure/layout

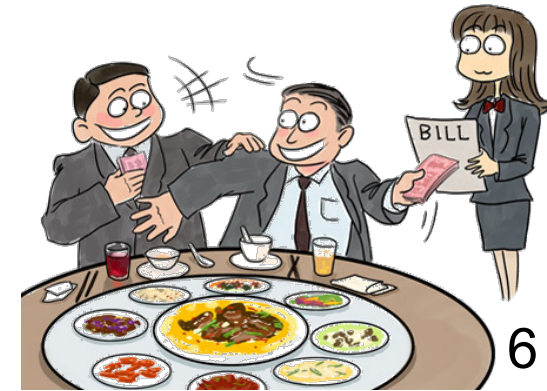


Match between system and the real world

The system should speak user's language, with pictures and concepts familiar to the user, rather than system-oriented terms



Adapt the software to the users' mental model



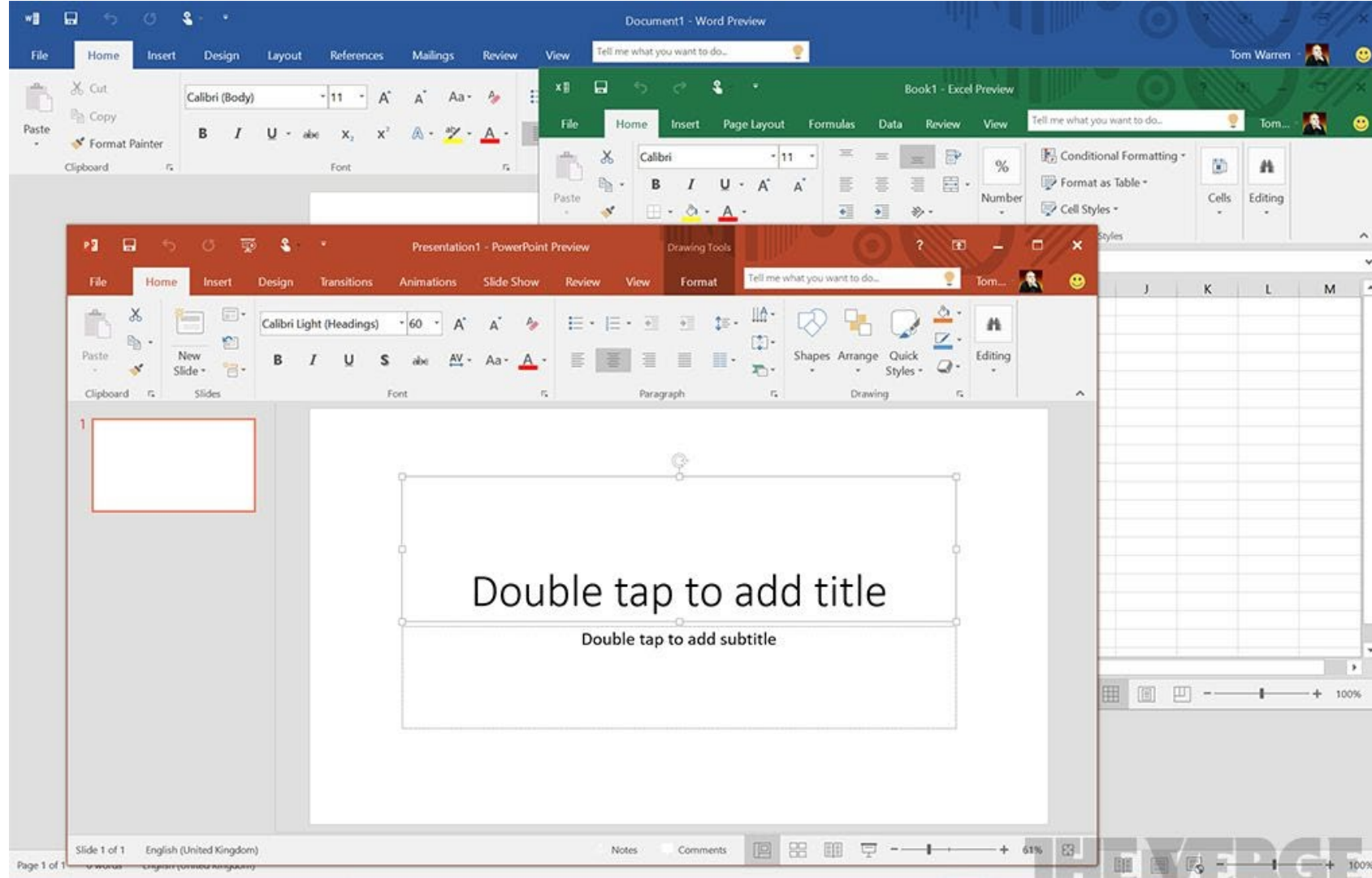
Consistency and standards

All similar words, situations, or actions should have the same style and meaning:

1. Consistency with other pages of the system
2. Consistency with similar or related systems

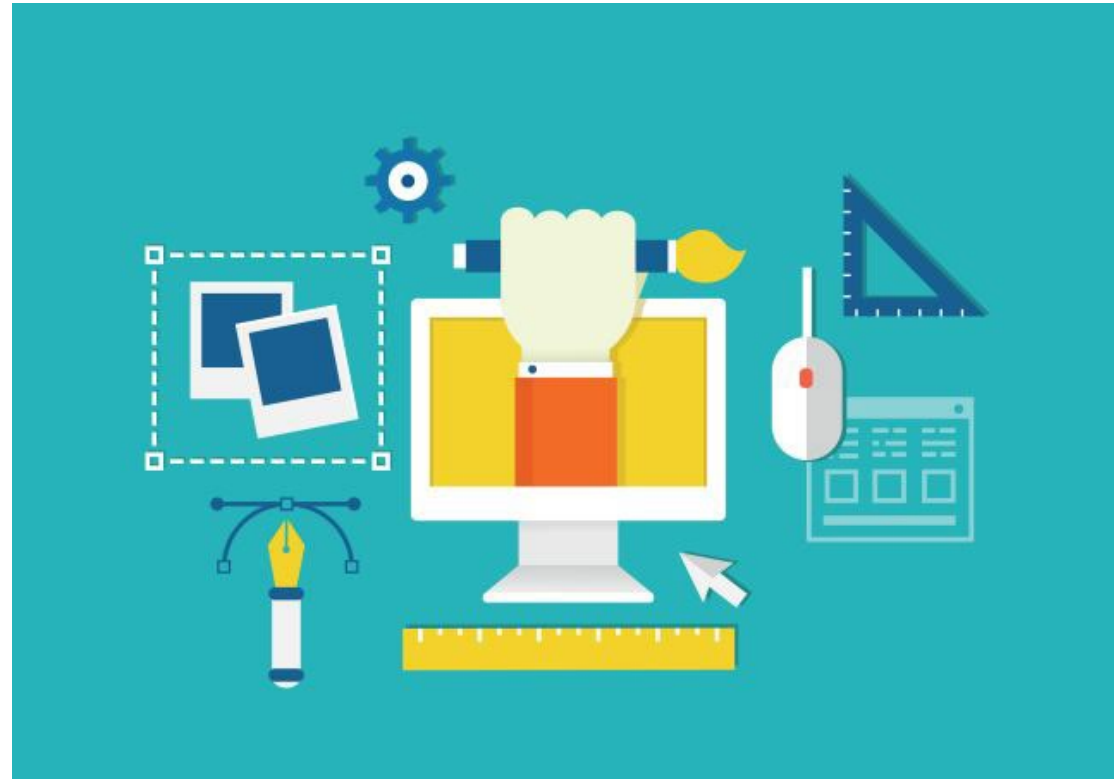


Consistency and standards

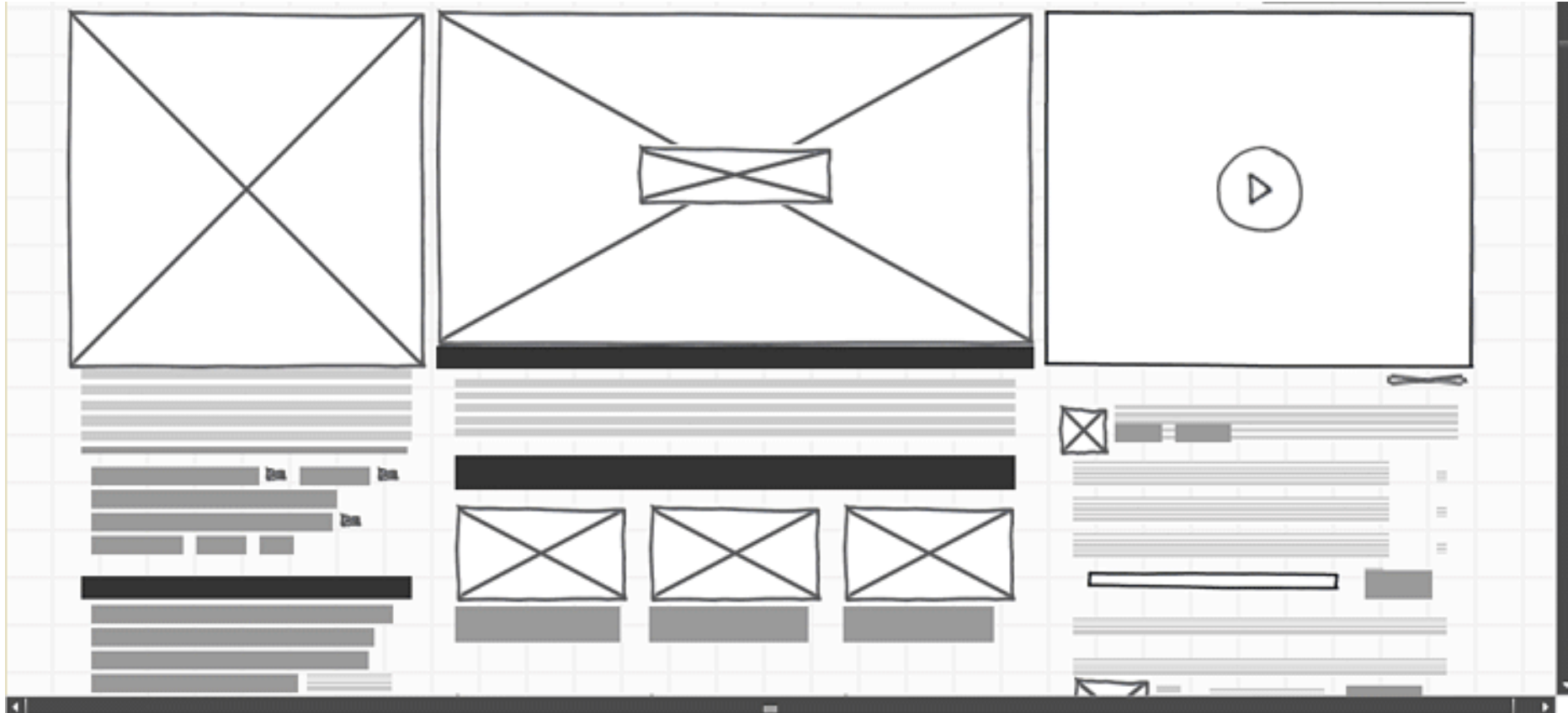


What does visual design mean?

Visual design focuses on the aesthetics of a site and its related materials by strategically implementing images, colours, fonts, and other visual elements



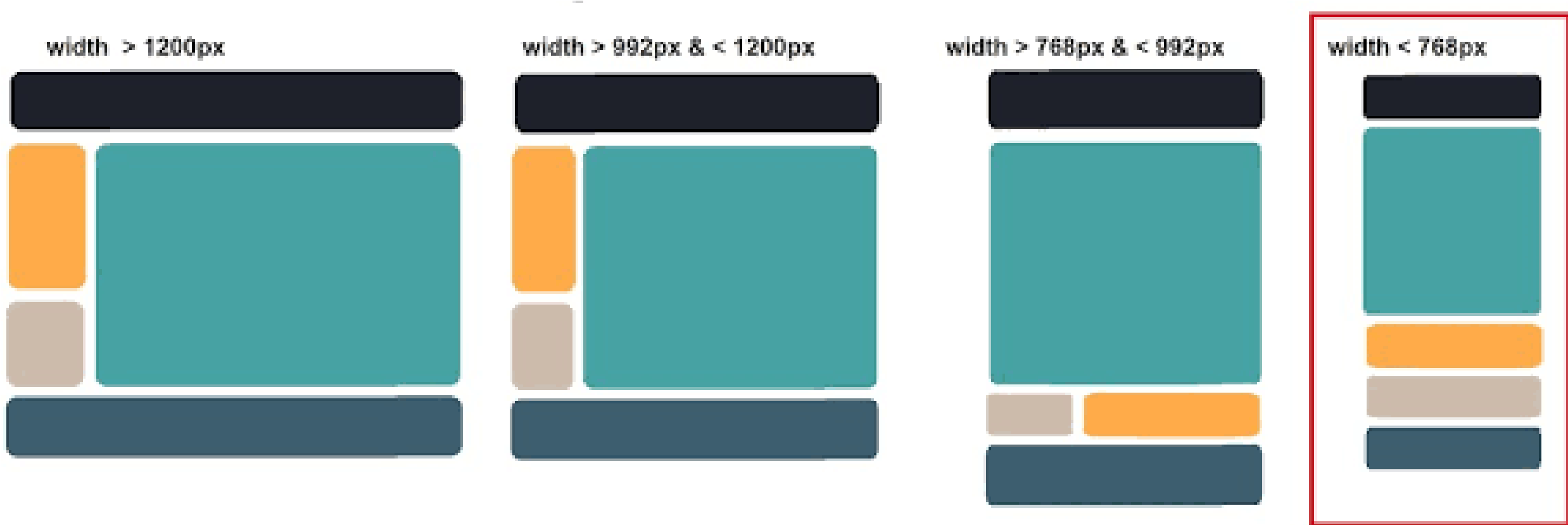
Designing the Layout



Layout Considerations

- Layout must have some relationship to the users' workflow
- Everything placed in the user's view creates something to think about, increasing the cognitive load
- Understand the tasks
- Study the ordering of the tasks
- Order the user interface elements so that it matches the workflow
- Consider visual groupings (reflect on the perceptual groupings)

Responsive / Adaptive Layout



Some Visual Interface Design Principles

- Avoid visual noise and clutter
- Redundant info uses up limited processing capacity
- Keep things simple
- Too much clutter increases search

White Space

- Space provides a separation between elements
- Helps reduce visual clutter
- Used to help organise and structure related items
- Can assist with balance and clarity

Figure-Ground Principle and Colour

Use colours to change the focus and thus what the user perceives

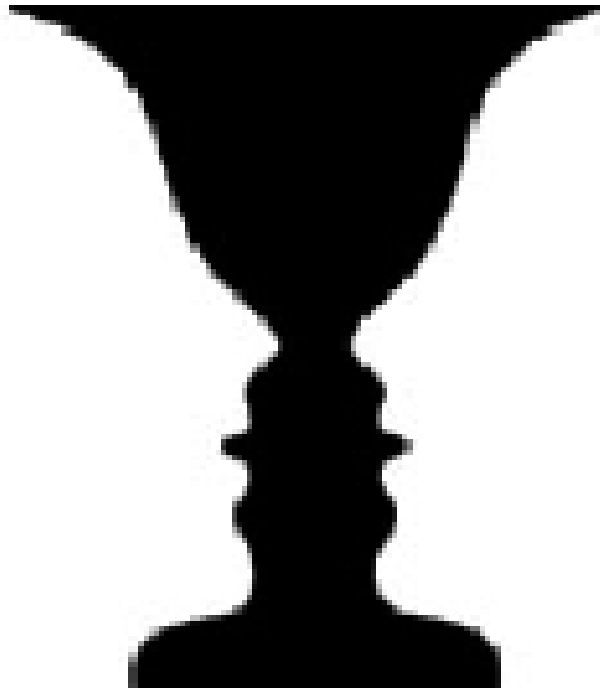


Image 1

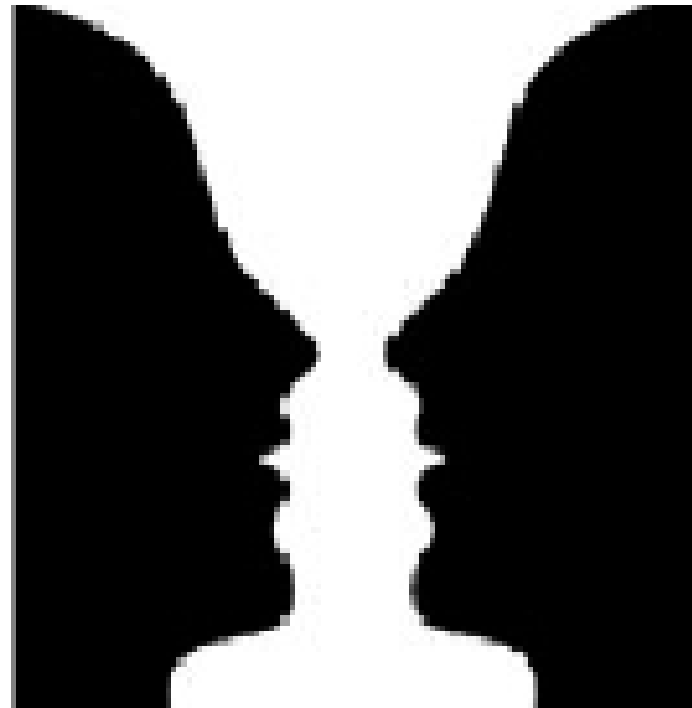


Image 2

Git and GitHub

Git

 **git** --fast-version-control

Git is a **free and open source** distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is **easy to learn** and has a **tiny footprint with lightning fast performance**. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like **cheap local branching**, convenient staging areas, and **multiple workflows**.





About

The advantages of Git compared to other source control systems.



Documentation

Command reference pages, Pro Git book content, videos and other material.



Downloads

GUI clients and binary releases for all major platforms.



Community

Get involved! Bug reporting, mailing list, chat, development and more.



Pro Git by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).



Latest source Release
2.42.0
[Release Notes \(2023-08-21\)](#)
[Download for Windows](#)

 [Windows GUIs](#)  [Tarballs](#)
 [Mac Build](#)  [Source Code](#)

Companies & Projects Using Git

 [About this site](#)
Patches, suggestions, and comments are welcome.

Git is a member of [Software Freedom Conservancy](#)

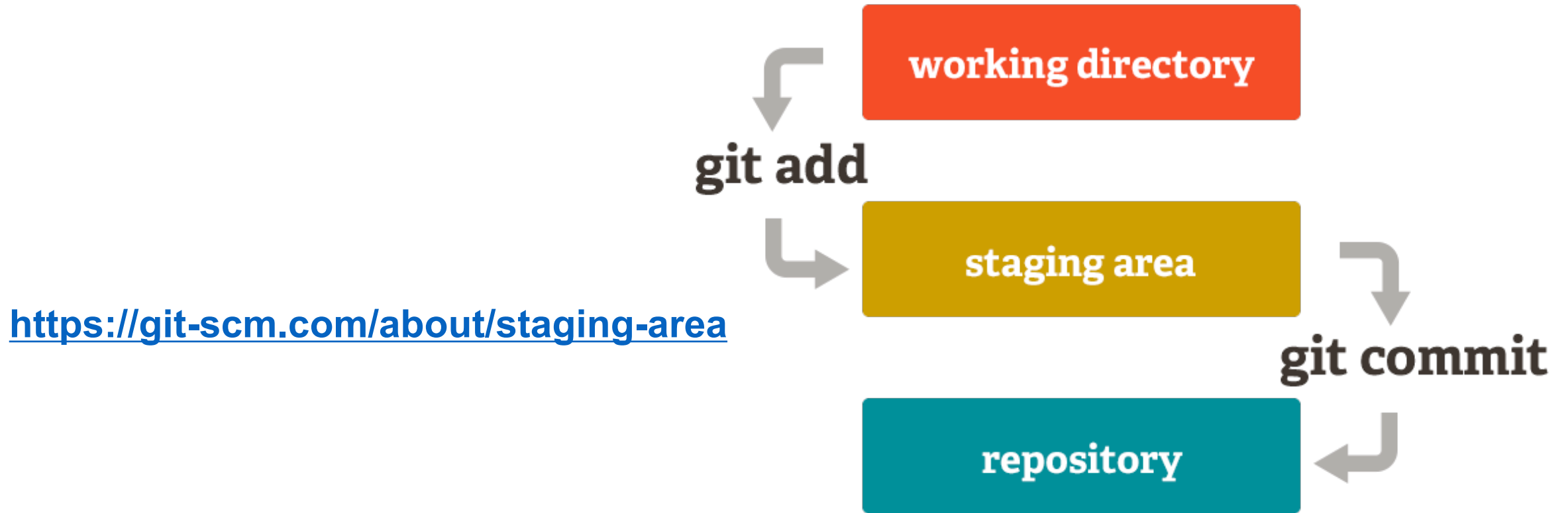
<https://git-scm.com>

What can we do with Git?

- Keep **history of changes** made to code/other resources
- Allow developers to **share** code
- Allow developers to **communicate change history**
- Maintain **multiple separate lines** of development for a **codebase**

Git

- Git is a **distributed Version Control System (VCS)** for tracking changes in computer files and coordinating work on those files among multiple users



Git

So why Git (and why VCS)?

- Our goals
 - **Share** code (or something else) **easily**
 - **Keep track** of any changes we make (and **undo** them with ease)
 - Maintain **multiple versions** of the same codebase
 - Clearly **communicate** what changes have been made
- Git is not like SVN (Subversion)
 - Git is **distributed** (but SVN is centralized)
 - Git is **powerful**
 - Git is **easier** than you think

More on Git

There are a lot of ways to learn more:

- [Git Ready](#)
- [Pro Git](#)
- [Git Community Book](#)
- [Git Website](#)
- [Man pages](#) (for reference)
- [Git Tutorial](#) - W3Schools

Try also:









- [Git Tutorials and Training | Atlassian Git Tutorial](#)



GitHub

- GitHub is a **provider of Internet hosting for software development** and **version control** using **Git**
- It offers the distributed version control and source code management (SCM) functionality of Git plus its own features
- It provides access control and several collaboration features such as bug tracking, feature requests, task management, continuous integration and wikis for every project

GitHub Classroom

GitHub Classroom





    classroom.github.com/classrooms/145270740-unsw-cse-comp3900-9900-23t3-2023term3     Paused

 Classroom GitHub Education 

Classrooms / unsw-cse-comp3900-9900-23T3-2023Term3


unsw-cse-comp3900-9900-23T3-2023Term3




unsw-cse-comp3900-9900-23T3



 Assignments **1**  Students **561**  TAs and Admins **15**  Settings

Assignments

+ New assignment

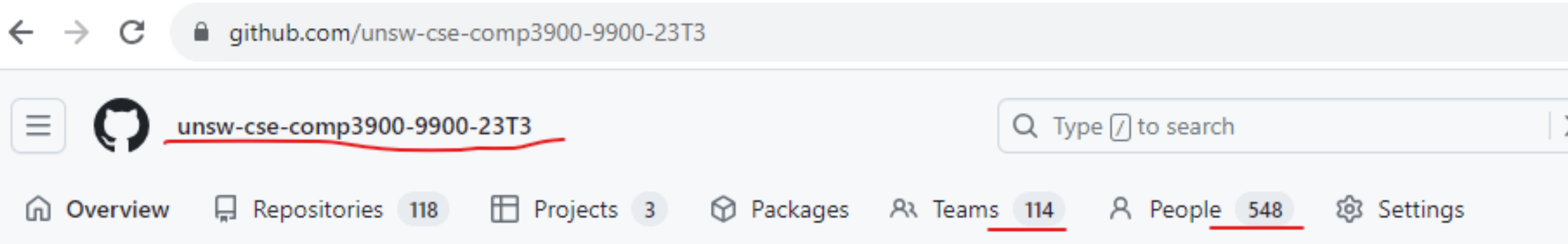
We value your feedback! Please take a few minutes to complete our [survey](#). 

capstone-project  Copy invite link  

 Active  Group assignment for COMP[39]900 23T3

GitHub Organisation

As of Tuesday 26/09/23 @ 11pm (Week 3)



28 students still did not accept the invite!

GitHub Classroom

As of Tuesday 26/09/23 @ 11pm (Week 3)

[Classrooms](#) / [unsw-cse-comp3900-9900-23T3-2023Term3](#) / [capstone-project](#)

capstone-project

👥 Group assignment ● Active

Assignment Details

Total teams 561

561 Rostered

37 Not on a team



Accepted teams 116

116 Teams

GitHub and Jira Integration

GitHub and Jira Integration

- GitHub.com + Jira Software integration
 - Connect your **code** in GitHub with your **project management** in Jira

<https://github.com/integrations/jira>

- Connect Jira Cloud to GitHub
 - When your GitHub account is linked to Jira Software, your team gets to see their **branches**, **commit** messages, **pull** requests, **builds** and **deployments** right in the context of the **Jira Software issues** they are working on

<https://support.atlassian.com/jira-cloud-administration/docs/integrate-with-github/>

GitHub and Jira Integration

The screenshot displays the Jira Software interface. At the top, the navigation bar includes 'Jira Software', 'Your work', 'Projects', 'Filters', 'Dashboards', 'Teams', 'Apps', and a 'Create' button. The 'Projects' tab is active, showing the breadcrumb 'Projects / 9900H15AperiPeri_Chicken'. The left sidebar contains a 'DEVELOPMENT' section with the 'Code' tab selected, indicated by a red arrow. Below this is an 'OPERATIONS' section with 'Deployments', 'Project pages', 'Add shortcut', and 'Project settings'. The main content area shows the 'Code' section for the project, with the heading 'Latest repositories' and a subtext 'The most recent repositories with commits, branches, or pull requests linked to this project.' A single repository is listed: 'capstone-project-9900h15...' with a GitHub icon and the text 'Last updated 4 minutes ago'. A red arrow points to the repository name, and another red arrow points to the 'GitHub' label above it. Below the repositories section is the 'Linked pull requests' section, which states 'Your team's linked pull requests from the last 30 days. [Learn more](#)'.

Project Proposal

Project Proposal

- The Project Proposal is due **Week 3 Friday @ 9pm**
- It is worth **10%** of the total marks for the course
- Teams are required to submit, at least, **10-page report** (not including the **title page**, the **table of contents**, and **references pages**) that:
 - **prioritises the project requirements**
 - develops an **execution plan** to achieve project goals

Background

- Worth **10%** (or **1 mark out 10**)
- Clearly identifies the **problem(s)** being solved
- Includes preliminaries of the problem **domain**
- Identifies, at least, **two existing related work or systems** in the same problem domain, and their **drawbacks**

User Stories and Sprints

- Worth **50%** (or **5 marks out of 10**)
- **Product backlog** of correctly structured **user stories**, describing the functionality to be delivered, with **screenshots** showing all these user stories defined in **Jira**
- The entire text of each user story should be **readable** inside the report (otherwise provided)
- The **start** and **end dates** for all (preferably **three**) sprints envisaged during the term are defined
- The **defined sprints** should allow the team to undertake a **progressive demos** in each of **Weeks 5 and 8** as well as the **final demo** in **Week 10**

User Stories and Sprints (cont'd)

- User stories in scope are identified for, at least, the **first sprint** with screenshots showing all user stories allocated to the first sprint in Jira
- The entire text of each user story is **readable** in the report (otherwise provided)
- The report **clearly** communicates how all project **requirements** are **satisfied** by the defined **user stories**
- The report also describes how some of the defined user stories to be implemented provide **novel** functionality **compared to existing related systems**

Sprint 1 Example

▼ MFS Sprint 1 19 Jun – 30 Jun (14 issues)

User registration page, user login page, password reset page, some part of main page (movie rating ranking, popular movies), movie feed function and movie detail page should be completed.

- 📌 IJ9900-4 As a registered user, I want to be able to log into the system with my own username and password so that only the authorized person and me can use the system. **USER STORIES**
- 📌 IJ9900-5 As a registered user, I want to be able to reset my password so that I can secure my account if my password is leaked or forgotten. **USER STORIES**
- 📌 IJ9900-16 As a general user, I want to see a movie rating rank on the home page so that I can pick and watch the top-rank movies. **USER STORIES**
- 📌 IJ9900-17 As a general user, I want to see the popular content in the feed section so that I can pick and watch the popular movie. **USER STORIES**
- 📌 IJ9900-18 As a registered user, I can filter my feed to quickly find the desired content so that I can directly find the movies that I am interested in. **USER STORIES**
- 📌 IJ9900-19 As a general user, I want to see the title and poster of the movie so that the movie might appeal to me and I can gain a gorgeous visual experience. **USER STORIES**
- 📌 IJ9900-20 As a general user, I want to obtain the actors', directors', and screenwriters' information and movie in the movie so that I can whether existing my favorite actors in this movie. **USER STORIES**
- 📌 IJ9900-21 As a general user, I want to see the movie tags and movie scenarios so that I can know further about this movie and determine whether I like this movie or not. **USER STORIES**
- 📌 IJ9900-22 As a general user, I want to see the year the movie was released so that I can know the approximate visual quality of the movie. **USER STORIES**
- 📌 IJ9900-23 As a general user, I want to see the score of the movie from the other popular movie review website, so that I can get a majority opinion. **USER STORIES**
- 📌 IJ9900-24 As a general user, I want to know the movie release date and the runtime of the movie, so that I can plan my time to watch the movie. **USER STORIES**
- 📌 IJ9900-25 As a general user, I want to be noticed the country and language of the movie, so that I can determine whether the script language is English or not. **USER STORIES**
- 📌 IJ9900-3 As a general user, I want to be able to register a unique account so that I can start using the system. **USER STORIES**
- 📌 IJ9900-26 As a general user, I want to know the general plot of the movie so that I can determine whether the plot of the movie is interesting. **USER STORIES**

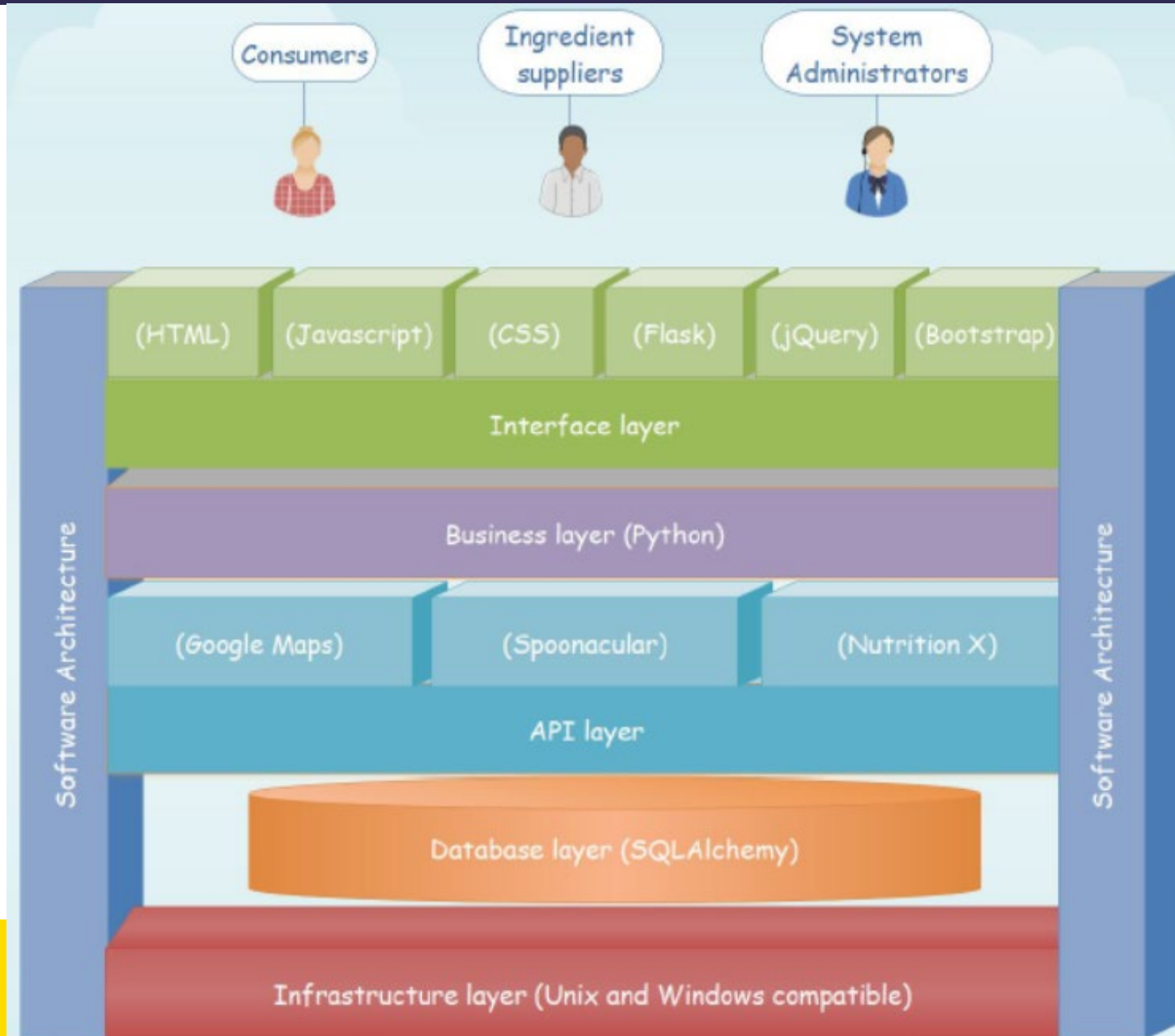
System Architecture

- Worth **15%** (or **1.5 marks out of 10**)
- A clear software architecture diagram showing, at least, the **presentation**, **business**, and **data** layers in the system, and what each layer contains
- A clear description of the **external actors** (i.e., **user types**) and how they interact with the system
- A clear description of the technologies/languages planned for use (e.g., MySQL, SQL Server, .NET, Java, and Python)
- The report also includes all third-party functionality planned to be used (e.g., clouds/services/APIs/libraries/code)

System Architecture (cont'd)

System Architecture Diagram Example

- **User types (must)**
- **Interface layer (must)**
- **Business layer (must)**
- API layer (optional)
- **Database layer (must)**
- Infrastructure layer (optional)

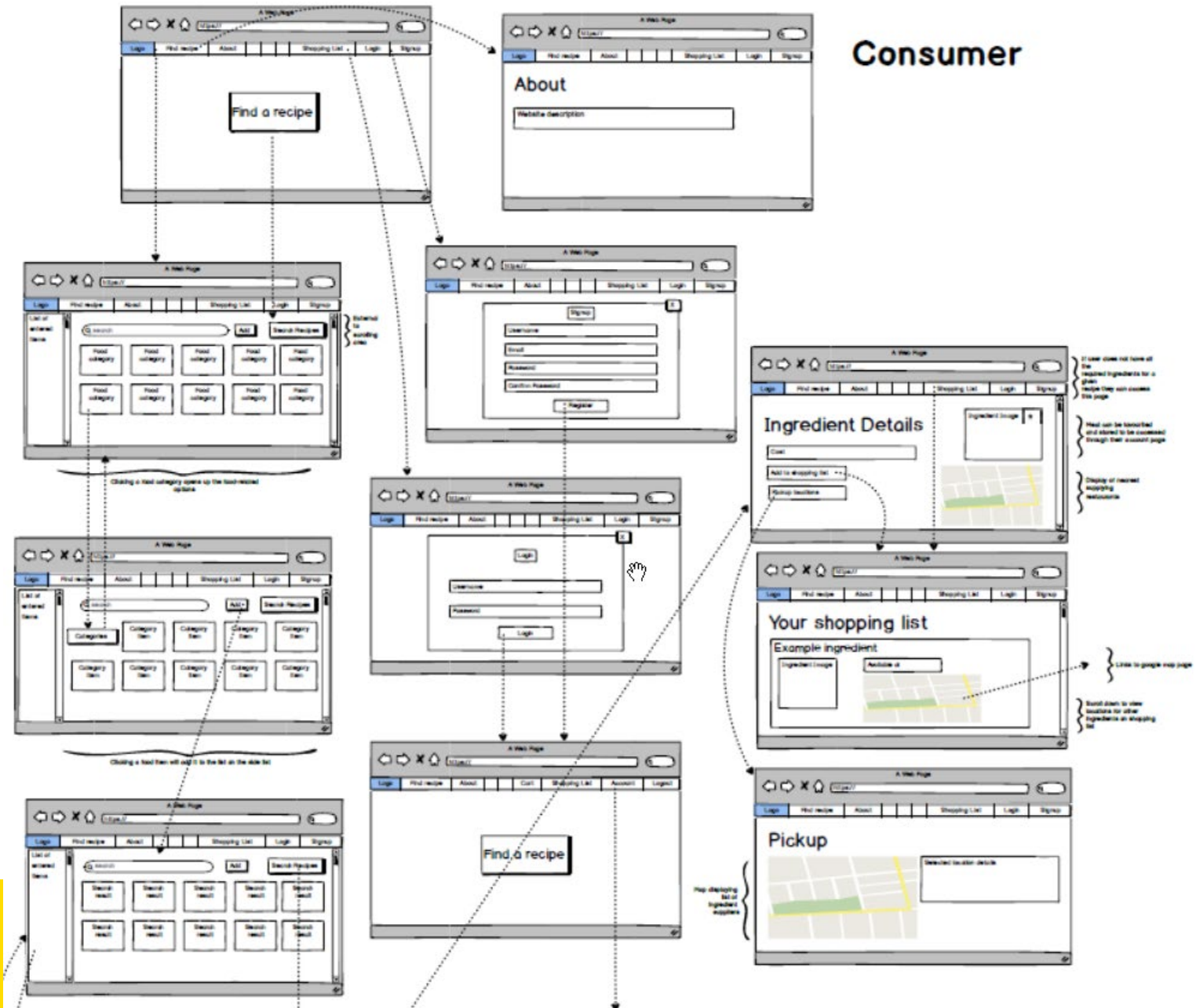


User Interface and Flow Diagrams

- Worth **20%** (or **2 marks out of 10**)
- **Storyboards** should be developed to illustrate the **system functionality** and how users interact with the system to be developed
- One **storyboard** can cover **multiple user stories**
- **All user stories** should be covered by these **storyboards**

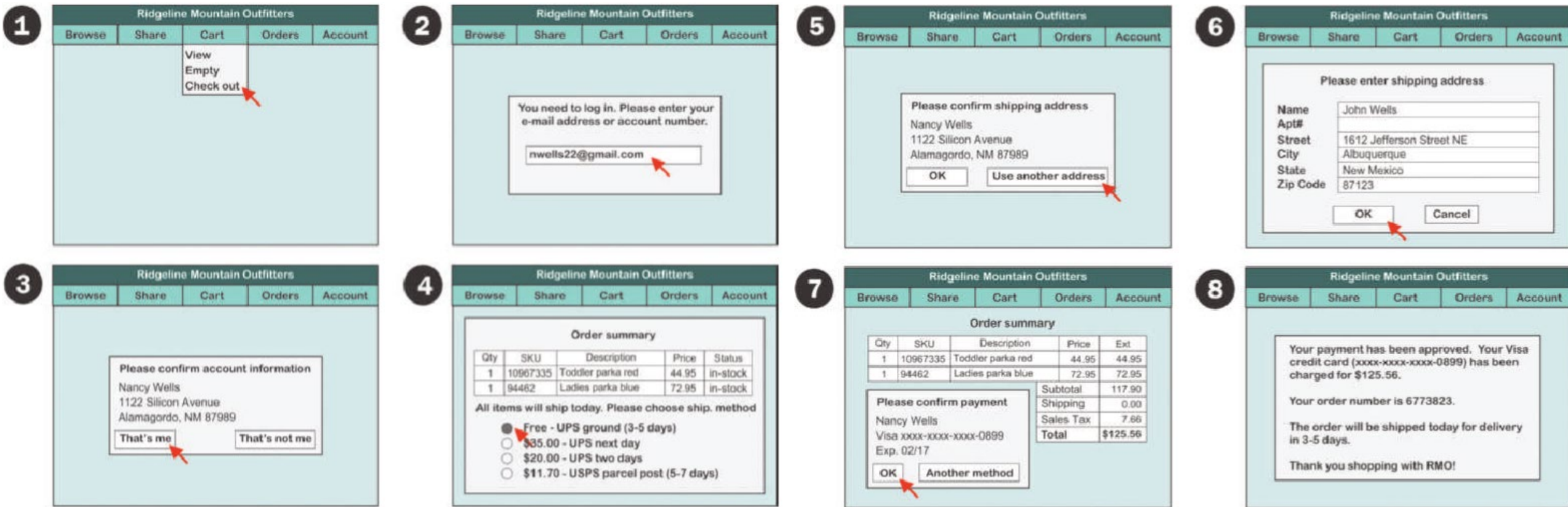
User Interface and Flow Diagram (cont'd)

Storyboards and Flow Diagram Example



User Interface and Flow Diagrams (cont'd)

Storyboard example for **Check Out Sopping Car** use case (user story)



Source: Satzinger, J. W., Jackson, R. B. & Burd, S. D. (2015). Systems Analysis and Design in a Changing World, 7th edition, Cengage Learning

Report Formatting

The Project Proposal should:

- Be self-contained (i.e., no content should be outside of the report and simply linked to)
- Include a **title page** containing **course code**, **course name**, **project title**, a **nominated group name**, **each member's details** (name, student ID, email, and role), and **proposal submission date**
- Be **at least ten (10) pages** long (at **most 12pt font** with reasonable margins and spacing), not including the **title page**, the **table of contents**, and **references** pages
- Be in **PDF** format
- Include a **table of contents** and **page numbers**
- Include **in-text citations** and **list of references** and use either **APA** or **Harvard** referencing style

Week 3 Lab Tasks

Week 3 Lab Tasks

- Your team is recorded in Moodle with a correct team name and all team members are included. This is very important for assessment submissions and marking
- Your team is signed up to Jira and has invited the tutor/mentor as site-admin. This is also very important since the mentor need to do the Jira-GitHub integration for you and mark the project proposal
- You accepted your invite and linked properly your zID with your GitHub account in the GitHub Organisation otherwise contact your mentor to fix your unlinked zID

Week 3 Lab Tasks (cont'd)

- In GitHub, you have a **team**, a **repository**, a **maintainer**, and a **folder/branch where individual work diaries are stored and updated regularly** (at least once a week)
- Your team met with the project clients at least once
- Your team started putting the Product backlog in Jira
- You are advised to have **three (3) sprints** as per the document "Suggested Sprints Structure" uploaded to Moodle under Week 2 lecture
- You are working on the the project proposal due **Week 3 Friday 29/9/23 @ 9pm**

Q & A