CSC290 Communication Skills for Computer Scientists

Lisa Zhang

Lecture 3; Jan 21, 2019

Announcements

- Critical review article #1 due Sunday 9pm
- Drop-ins available at the RGASC Wednesday Jan 24th 1pm-3pm
- Submit on Quercus (not MarkUs)

Project Managment

Why?

This course is about communication skills for computer scientists, not project managers. Why are we talking about project management?

Why?

This course is about communication skills for computer scientists, not project managers. Why are we talking about project management?

- You will be a part of software projects.
- You will work with other developers.
- You will use project management tools.
- You will work with project managers.
- You may even choose to become a project manager.

Your Group Project

To understand the aspects of communication in software projects, you will be working in a group to write a small piece of software.

Project Management

A Project Manager (PM) is a professional who plans, procures, and manages the execution of projects

- "Governing body" = Project Management Institute https://www.pmi.org/
- Project Management Body of Knowledge (PMBOK)
- Certification: Project Management Professional (PMP) and many others

We'll talk about project management from a software perspective.

Most Software Projects Fail

Project management is hard!

According to a report by The Standish group:

- About 30% of software projects will be canceled before they're completed.
- Over 50% of software projects will run over budget by nearly twice as much as originally budgeted.

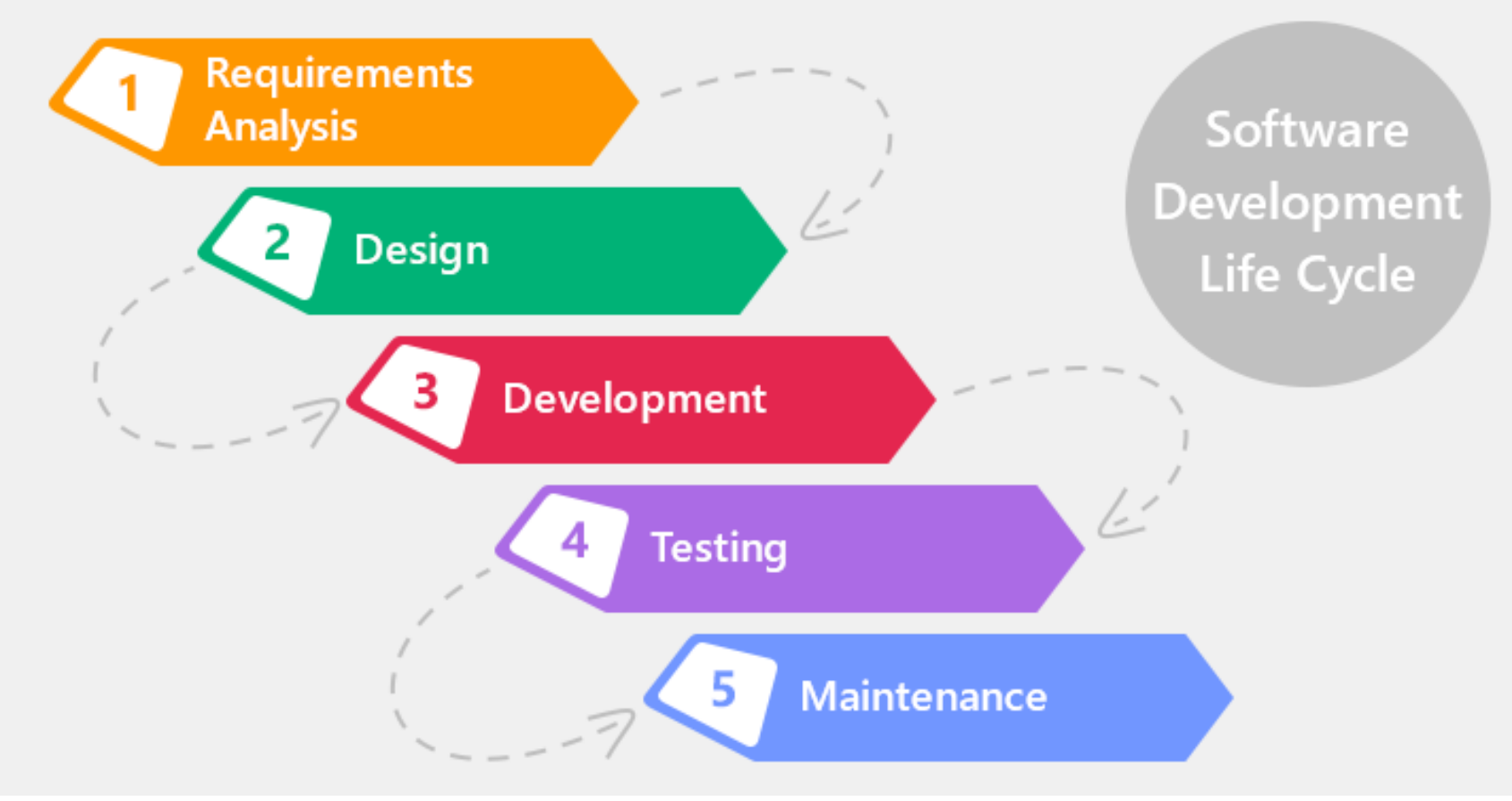
https://www.projectsmart.co.uk/white-papers/chaos-report.pdf

Why do softare projects fail so often?

- Unrealistic or unarticulated project goals
- Inaccurate estimates of needed resources
- Badly defined system requirements
- Poor reporting of the project's status
- Unmanaged risks
- Poor communication among customers, developers, and users
- . . . and more

From https://spectrum.ieee.org/computing/software/why-software-fails/3

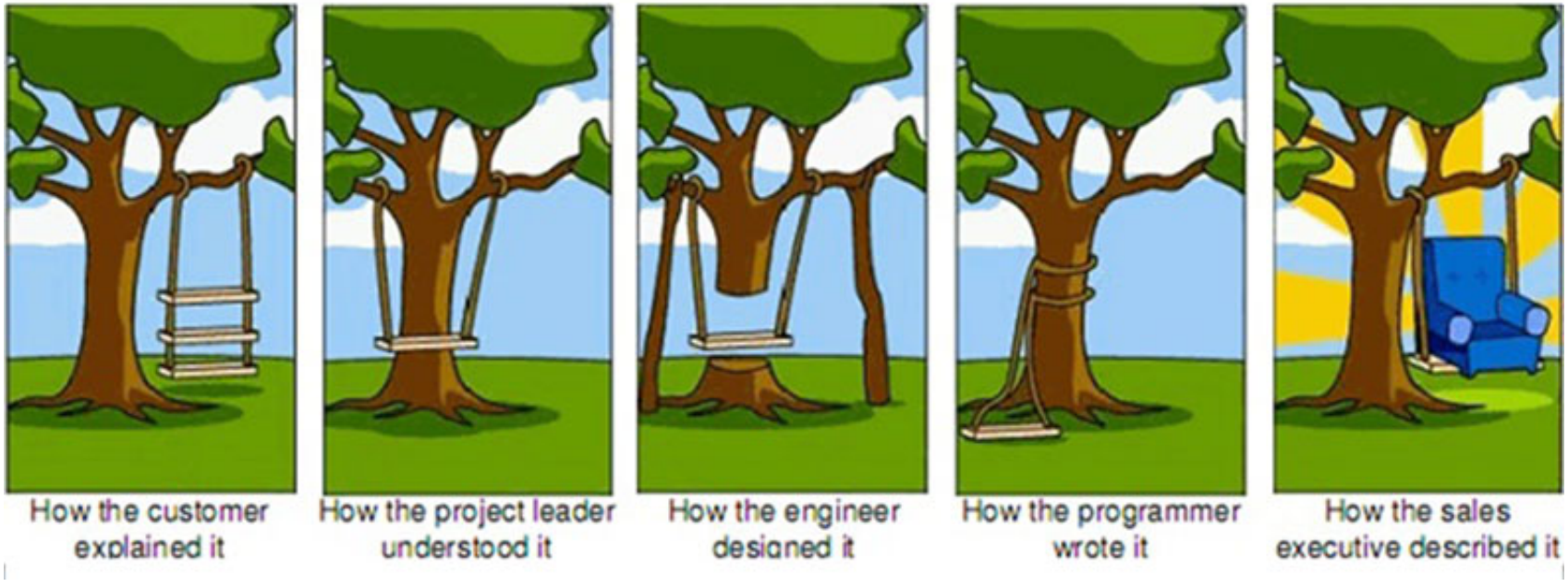
Software Development Lifecycle



We'll focus on steps 1, 2 today, and step 3 in a future week.

Requirement Analysis

Why Analyze Requirements?



Goals

- Identify **stakeholders**
- Elicit **requirements**
- Determine **deliverables**
- Determine the **scope** of the project
- Estimate **time and resource** use
- Choose **milestones**
- Identify **risks**

Stakeholder

Person or party with a interest in the project/process; can affect or be affected by project/process.

Stakeholders help define their requirements of the project.

Stakeholder

Person or party with a interest in the project/process; can affect or be affected by project/process.

Stakeholders help define their requirements of the project.

Example: UofT just replaced its Learning Management System. Who were the stakeholders include:

Stakeholder

Person or party with a interest in the project/process; can affect or be affected by project/process.

Stakeholders help define their requirements of the project.

Example: UofT just replaced its Learning Management System. Who were the stakeholders include:

- students
- instructors
- teaching assistants
- support staff
- university administration

Non-Stakeholders

- The developers and competitors are **not** stakeholders!
- They are not the *users* of the resulting product or service, and do not help define the requirements of the project.
- The maintainers of a project can be a stakeholder.

Requirement Elicitation

- "Elicit" defined as:
  - Draw forth or bring out (something latent or potential)
  - Call forth or draw out (as information or a response)
- Engage stakeholders actively to define requirements that are complete, clear, correct, and consistent

Requirement Elicitation

- "Elicit" defined as:
  - Draw forth or bring out (something latent or potential)
  - Call forth or draw out (as information or a response)
- Engage stakeholders actively to define requirements that are complete, clear, correct, and consistent

Example: You are building a website for a dentist, what are some possible requirements?

Software Requirements

- **Functional requirements**: defines the *behaviour* of the software
- **Non-functional requirements**: judges the *operation* of a system
  - Performance
  - Reliability
  - Availability
  - Security
  - Maintainability
  - Portability

Deliverables

A tangible or intangible product or service produced.

Deliverables

A tangible or intangible product or service produced.

Example: You are building a website for a dentist, what are some possible deliverables?

Deliverables

A tangible or intangible product or service produced.

Example: You are building a website for a dentist, what are some possible deliverables?

- project proposal (or estimate)
- project plan
- website mock up
- website design
- the landing page
- the entire website

What's In Scope? What's Out of Scope?

Features that are **in scope** are part of the project.

Features that are **out of scope** are explicitly excluded from the project.

It is important to agree ahead of time what features will be in scope and out of scope.

What's In Scope? What's Out of Scope?

Features that are **in scope** are part of the project.

Features that are **out of scope** are explicitly excluded from the project.

It is important to agree ahead of time what features will be in scope and out of scope.

Example: What might a dentist want in his/her website that is time-consuming to build?

Estimating time and effort

In order to estimate the resources required for the project, we break down the project into parts.

We build a **work breakdown structure**, which organizes the team's work into manageable tasks.

Work Breakdown Structure

Hierarchical description of all of the tasks in the project.

Useful for planning and reporting the status of the project.

Example

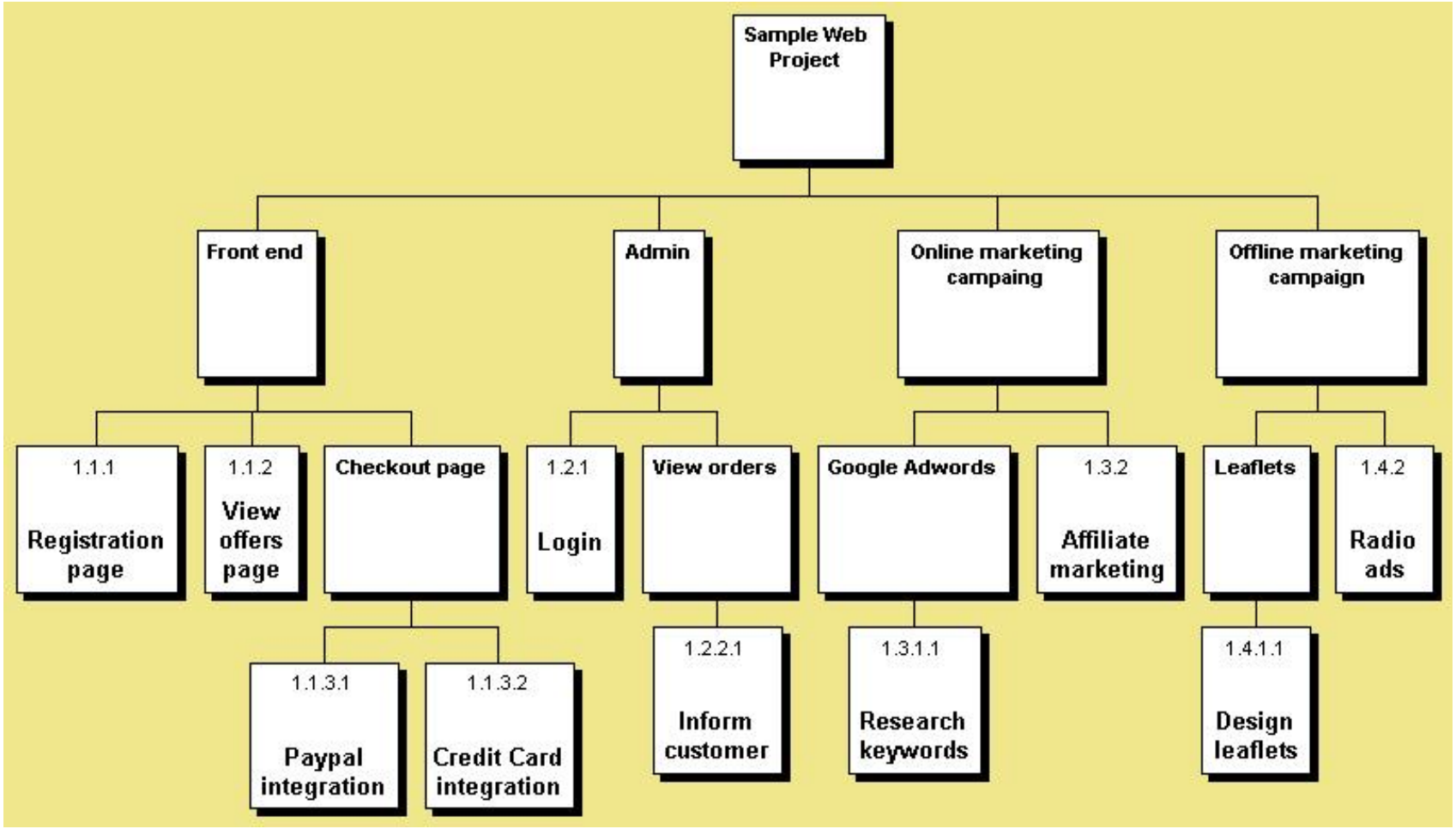


Figure 1: from https://project-management.com/

WBS Tasks

Each task should...

- have a defined start and end, and associated deliverable.
- doable without interruption (e.g. waiting for another task).
- be small enough so you can estimate its time and cost.

You should be able to identify the task status at any point.

Alternate Format

Level 1	Level 2	Level 3
1 Foundation	1.1 Excavate	1.1.1 Dig 1.1.2 Level
	1.2 Frame	
	1.3 Concrete	1.3.1 Pour 1.3.2 Cure
2 Exterior		
3 Interior		

Why?

- It is easier to estimate how long small tasks will take.
- It is easy to underestimate how long large tasks can take.

Gantt Chart

A Gantt Chart also takes into account task dependencies.

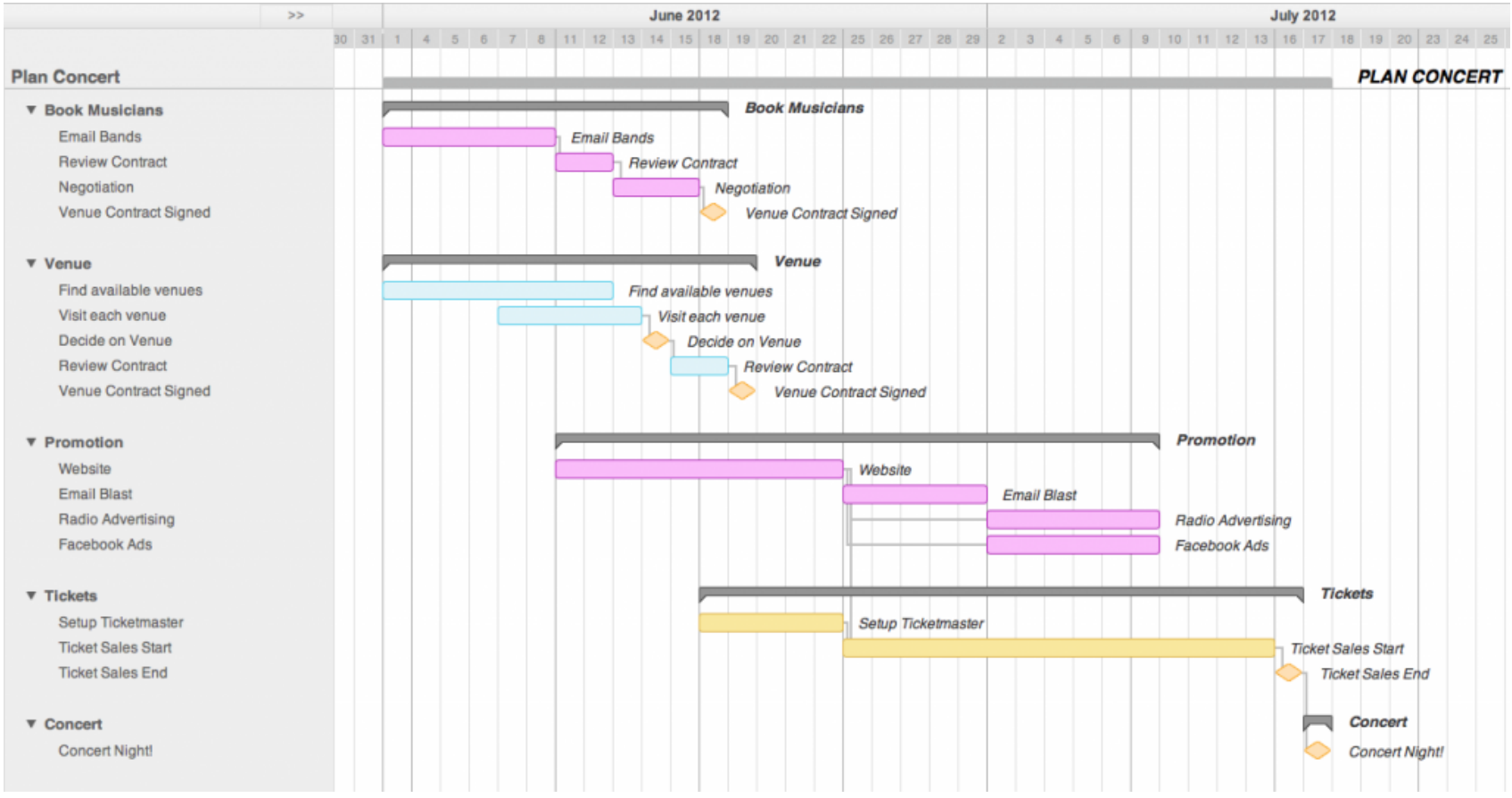


Figure 2: from http://executivepropmgmt.co/

Identifying milestones

A **milestone** is a significant checkpoint in project timeline.

The nature of the project might change before and after a milestone.

Example: You are building a website for a dentist, what are some possible milestones?

Identify Risks

- Anticipate possible problems that may occur
- Plan for possible responses to such problems
- For example, using a Risk Register
  - https://en.wikipedia.org/wiki/Risk_register#Example

Project Documents

- Project Proposal: to gain approval of stakeholders
- Project Charter: agreement with stakeholders
- **Project Plan**: to align people working on the project

They each have different goals/audiences, but are similar in that they list the project goals, stakeholders, scope, deliverables, resources, milestones, and risks.

Your Group Project

Your first deliverable is a **project plan**.

These are some project charters to help you:

[1] https://en.wikipedia.org/wiki/Project_plan
[2] http://www.pmsouth.com/wp-content/uploads/2014/07/project_charter_template.pdf
[3] http://s.casual.pm.s3.amazonaws.com/toolkit/WebsiteDesign.pdf
[4] http://isoconsultantpune.com/wp-content/uploads/2015/05/MassCommunicationProjectCharter.pdf