

Software Engineering Project Workshop (SENG202)

Matthias Galster

Phase 1 – tasks (part 4)

July 16, 2020

Deliverables

- Project setup checklist
- Design document
- Reflections and logs
- Presentation



Content of design document

Executive summary

1. Business and system context
2. Stakeholders and requirements
3. Acceptance tests
4. GUI prototypes
5. Deployment model
6. Detailed UML class diagram
7. Risk assessment
8. Project plan

References

Appendix

Risks + risk assessment

ID	Description	Impact	Likelihood	Responsibility	Consequences	Prevention
		1 - 3	1 - 3	"Owner"		


- Different types of risks
 - [Business]
 - Technical
 - Implementation
 - Operational
 - Etc.

Risk assessment

- Analyze impact
 - Low
 - Medium
 - High
- Analyze likelihood that risk becomes a problem
 - Low
 - Medium
 - High
- Include “owner” of risk, how risk can be mitigated, and what should be done when risk becomes a problem

Example (cost of unplanned delay)

- If customer doesn't like our form layouts then there may not be time to implement a new one before the release goes live unless we drop a feature.
 - Likelihood: 10%
 - Impact: 3 staff-days
 - “Exposure” (likelihood x impact): 0.3 staff days
- Prevention
 - Show first form implemented to customer
 - Develop two versions
 - Implement switchable layouts
 - Implement layout editor
 - Etc.



Compare cost of prevention strategy to exposure

Examples

- Team conflicts
- Unfamiliar development tools
- Unfamiliar API's / libraries, various programming skill levels
- Change in team composition
- Miscommunication with lecturers
- Loss of data, problem with import of data
- Inaccurate calculations, analyses
- Product does not agree with stakeholder expectations
- Code written by individual team members not readable by others
- GitLab, Google Drive, etc. become unavailable
- No internet

Content of design document

Executive summary

1. Business and system context
2. Stakeholders and requirements
3. Acceptance tests
4. GUI prototypes
5. Deployment model
6. Detailed UML class diagram
7. Risk assessment
8. **Project plan**

References

Appendix

Deliver project that is...

... on time

... on budget

... meet requirements

... of adequate quality

... sustainable



Notes for project plan in SENG202 (1)

- Based on outline, dates, tasks, the project plan should include
 - Deliverables (i.e., the end of each phase) as **major milestones**
 - Own milestones as **minor milestones**
 - E.g., for the first phase, drafts of the different parts of deliverable
 - Doc/code freeze
 - Responsibilities
- Consider **other commitments of team members**
 - E.g., assignments, tests or exams, or commitments outside university

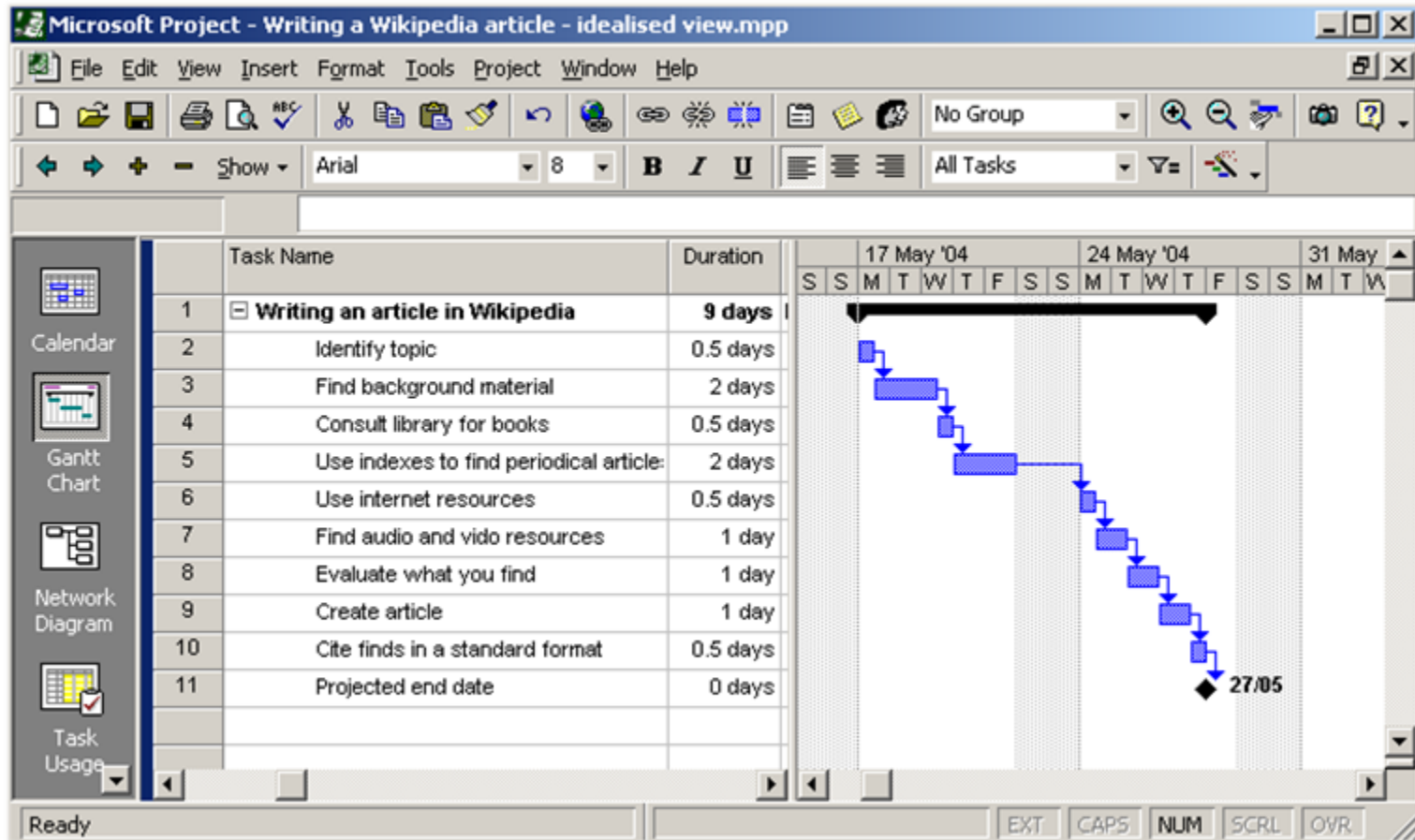
Notes for project plan in SENG202 (2)

- Make sure to include some **buffers** in your project plan
 - Delays and unexpected difficulties will most likely occur
 - Effort estimated for different activities should be conservative
 - The unexpected always happens – always allow contingency
- **Informal checklists** that complement your project plan may help you keep track of individual tasks
 - Requires measurable targets
 - Make them visible to whole team
- Project plan may be **refined later** once you have a clearer idea about your systems

Notes for project plan in SENG202 (3)

- Use some **tool support**
 - For project planning and tracking your progress
 - Chosen tool would depend on your preference
 - Many industrial projects use Microsoft Project
 - Create an Excel / Google spreadsheet with project dates, deliverables, etc.
 - Create Google Calendar
 - Other tools, e.g., Trello
 - GanttProject (<http://www.ganttproject.biz/>)
 - **Whiteboards, sticky notes**
 - At a later stage: consider using some bug tracking software

Gantt chart in MS Project – example



- Tasks could (and most likely will) overlap
- Free alternative: GanttProject (<http://www.ganttproject.biz/>)

Task duration and dependencies

Milestone	Date	Description	Related tasks
M1		Requirements document	T1 – T3
...	

Task ID	Task name	Days	Dependencies
T1	Define vision	2	
T2	Identify stakeholders	2	T1
T3	Identify concerns	1	T1, T2
...

Note:

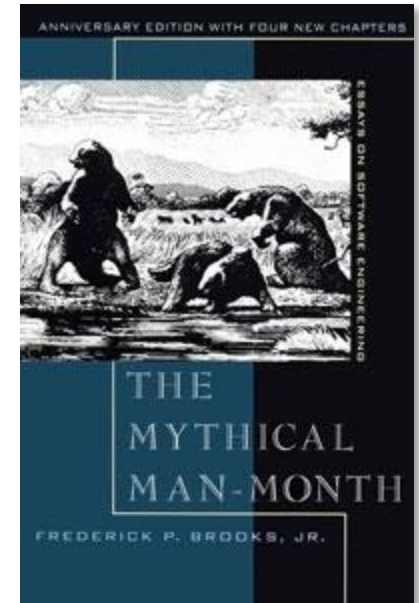
- Things will change (but you have to start somewhere)
- Focus on major tasks
- Helps understand whether what you want to do is feasibility
- Things will change

Project scheduling – basic principles

- Split project into tasks and **estimate time / resources** for tasks
- Organize tasks **concurrently** to make optimal use of workforce
 - Productivity is not proportional to number of people working on task
- Minimize task dependencies to avoid delays caused by one task
- Measurable targets and milestones (end-points of activities)
 - **Good:** build prototype chat with RMI
 - **Bad:** read about RMI

**Project activities must produce tangible outputs
and help judge project progress**

Brook's Law



Hofstadter's Law

“It always takes longer than you expect,
even when you take into account
Hofstadter's Law.”

Advice for project planning

- As you go along in SENG202, regularly perform the following:
 - Review – **what has been done**
 - Project progress, project
 - Feedback
 - Adjustment of plan
 - Retrospective – **increase efficiency and effectiveness**
 - What went well?
 - What went wrong?
 - What can we improve?
 - Planning – **what will be done**
 - What, who, when
 - Outline of plan (e.g., for next week)
 - Define measurable targets

Weekly 5-minutes “stand-ups” in labs

Hint



The project plan is a rough outline of what tasks need to be completed when and by whom, and what deliverables need to be produced. The project plan together with initial requirements helps you evaluate the feasibility of implementing your system in the available timeframe. The project plan should not only include course deliverables as major milestones, but also intermediate deliverables and milestones set by teams (e.g., status presentations, sections of reports, implementation of features). The project plan will be influenced by risks and should therefore include some generous time buffers to account for delays. The project plan should take other commitments from all team members into account (e.g., other courses, commitments outside university). Furthermore, effort estimates for different activities should be conservative. The project plan can be refined as you progress with your project.

Hint



Use tools to support your project planning and tracking progress. The chosen tool would depend on your preference. Many industrial projects use Microsoft Project. However, since this is a commercial product, you may not have access to it. Also, it is quite rich in features and thus may exceed the needs for your project. If you really want to use Microsoft Project then check out <http://www.cosc.canterbury.ac.nz/help/ELMS/index.shtml> for how to get it and the various other things available under our academic license programme. A free alternative is <http://www.ganttproject.biz>. Creating an Excel / Google spreadsheet with project dates, deliverables, etc. or free web-based tools like Trello may be the easiest option.

Looking ahead... the presentation

- At the end of Phase 1 (Deliverable 1)
- Content
 - What is the system context and business vision/rationale
 - Who are your stakeholders and concerns
 - What are main use cases (no detailed requirements), with GUI prototypes
 - What are the key drivers/quality requirements
 - Example (major) acceptance tests (optional)
 - Overview of UML class diagram (patterns, layers, packages, etc.)
 - What are main risks (what could go wrong; how is it mitigated)
 - Overview of project plan, how will goal be achieved (optional)

Reminder

Executive summary

1. Business and system context
2. Stakeholders and requirements
3. Acceptance tests
4. GUI prototypes
5. Deployment model
6. Detailed UML class diagram
7. Risk assessment
8. Project plan

References

Appendix

Documents are “linear”,
how you work is not

