System Anaylsis

Anna Pedersen

W1 - W4

1 Project management steps

- planning
  - The main purpose is to plan time, cost and resources for the project or problem
  - The planning is generally a document that outlines more details about the project, as well as provides a roadmap
- Scheduling
  - Scheduling allows us to plan each step of the project along with the expected time
  - We often use Gantt Charts and PERT chart to outline the time taken for each step of the project
  - Scheduling allows us to work out the critical path for the project timeline
- Budgeting
  - Budgeting helps us to allocate the financial resources of a project
  - This includes ensuring the budget can accommodate each step of the project without spending more than the project is worth.
- Tracking
  - Tracking allows us to keep up to date with every aspect of the project
  - Its important to keep an eye on each phase of the project to ensure its going to plan and there are no problems
  - Various tracking and collaboration software is avaliable for groups working on projects, such as BaseCamp

2 Linear - Cascade/Waterfall

- The waterfall / cascade model is a sequential design process in which progress is seen flowing down through the stages, like a waterfall
- Each phase must be completed before the next phase can begin, hense being easy and simple to understand
- A review is conducted at the end of each phase to make sure the project os being carried out correctly
- **Advantages**
  - Very simple and easy to understand
  - Phases do not overlap which allows us to prioritise each phase as it happens

  - Great for smaller projects where the requirements are well understood
  - what is proposed is what is expected

- **Disadvantages**
  - Once in the testing stage, going back is virtually impossible if it was not thought out in the concept stage
  - High amounts of risk and uncertainty due to large amount of planning
  - Poor model for larger and ongoing projects
  - Not suitable for projects that are likely to change

3 Iterative - RAD

- Rapid application development (RAD) is somtimes used as a general term to describe an alternative to the waterfall method
- It can also be used to describe a methodology created by James Martin
- In this methodolgy less empahasis is put on planning and more on development
- Components (or functions) are developed simultaneously as if they were mini projects
- Each component is given a deadline after which all components are gathered and made in to a working prototype
- The prototype is used to assess the users feedback regarding requirements and expectations from the project.
- **Advantages**
  - Significantly reduced development time
  - Encourages customer feedback
  - Increases re-usability of components in the system
- **Disadvantages**
  - Requires highly skilled designers and developers
  - High dependency on modelling skills
  - Depends on a strong team to identify requirements
- **When would you use RAD?**
  - When we need to have a system built in a short time frame
  - When we have a high avaliability of designers that are able to design the product and ....
  - When our budge is high enough to supply numerous deigners and tools needs needed for automated code generation.

4 Stages of the SDLC

- **What is the SDLC?**
  - The system development life cycle (SDLC) is a process commonly used for planning, creating, testing and deploying an information system
  - It can apply to software, hardware or a combination of both
  - The SDLC is defined by a number of clearly grouped activities, known as phases used to develop a finished project or product.
- **Why do we use the SDLC**
  - Gives us a structured, easy to follow approach when developing an information system.
  - Phases allow us to break down each step in the development and allows for less errors or discrepancies due to planning.
- **Preliminary Analysis**
  - first we define the problem of the system. Once complete we can allocate resources and prioritize tasks
    - What is the common goal? What are the equipment costs? How many employees do we have? Do we need more? What are the employees willingness to learn? What is the employee current model?
  - Feasibility report to test if the project can be done
    - A feasability study is getting your idea and breaking it down into quantifiable terms so you can see if it is worth undertaking or not
    - It includes technical feasilbility, if it is operational, if it is economically sustanable and if it fits within the schedule
    - Technical, operational, economic, scheduling
      - TOES
      - Economic → How much will it cost? How much needs to be spent? Break down each component, staff salary and you tryto make the most in depth and apprx cost you can at this stage
      - Schedule → How much time will it take? How much time do you have as a company? Our competitors coming out with a similar product before us.
      - Technical Feasibility → What technology will be used? Does it exist? Do we have it in our organisation? Do we need to develop anything to work in conjunction with this technology?
      - Operational feasability → What are the specific skills our staff need to operate this new system? Can our staff operate this new system or does it have new features they are unfamiliar with? Will we need additional training or outsourcing? Will we need to hire more staff?
  - Analysis
    - we define the project goals based off the client or end-users needs
    - Work out a model for the current system. We create buisness rules based on user needs, ER diagram can start here, normalisation to search out redundancy issues
    - We find out the requirements of the new system
    - Determine the cause of the problem
      - create a context diagram to see what interacts with the system
      - Create a Data Flow Diagram to pin point where the inforamtion is travelling throughout the system and how this could be more efficient/upgraded

  - Systems Design
    - Designing a newer and better system after analysing which improvements should be made
    - Logical deisgn → This explains what the new system will do
    - Physical deisgn → lists the equpiment needed to perform the logical design.
- **Design**
  - We design both the physical and logical parts of the system
    - logical → an abstract design usually by modelling, ERD
    - physical → technology specific details from which all programming and system construction can be accomplished
  - Creating a design which statisfys the application requirments
  - Changing from ”what” questions to how questions
  - Ensuring all specified functions are added to the system
  - Planning of the system documentation
  - Design GUI standards
  - design system architecture
  - design software components
  - Construct design prototype
  - Finalise testing strategy
  - Finalize conversion strategy
  - Ensuring design specifications are agreed upon and ready to develop
- Development
  - we gather our resources and build and test the system
  - the resources we obtain are the hardware and software that is needed
  - the system is created and tested
  - Design has laid the foundation for system development; the following phases ensure that the product functions as required
    - Establish standards
    - Hardware Acquisions
    - Combinging and intergrating small systems into the larger overall system and testing to ensure everythign is interoperable
    - Software completed
    - Ensure design specifications have been converted into a working inofrmation system that addresses all documented system requirements.
- Implementation
  - we implement the new system into its enviroment.
  - This can be done in many ways, each with their own advantages and disadvtanges.
  - Direct Cutover - Is a direct approach where the old system is cut and over written by the new system. The direct cutover appraoch causes the changeover from the old system to the new system to occur immediatly when the new system becomes operational
  - Pilot → A select group are given access to trial the new system before implemtnting globally
  - Parallel → both systems are run simultaneously until the new one is considered stable and usable
  - Phased → Parts of the new system are implemented as seen fit until the new system is fully installed.
  - System changover is the process of putting the new information system online and retring the old system. The four system changeover approaches description, advantages, disadvantages and the implications of using each of these approaches.

  - Evaluation and maintenance
    - We consider the performance of the system to make sure the new system is working and fits the requirements from the pre-analysis and analysis stages
    - Performance evaluation
    - We conduct fault finding and make corrections.

5 Data Gathering techniques

Resource

- Observation
  - When we simply look at the existing system and take notes on how it functions
- Questionnaire
  - We can create surveys or questionaries to give to users of the system to get some real time feedback and information regarding the system.
  - short survey questions to people who use the system will get lots of responses but wont be in depth.
- Interview
  - We can also interview existing users to examine their knowledge of the current system and also to gather feedback on potential changes or upgrades
  - Interview people within the system, eg. employees ,to get a detailed sense of the strenghs and weaknesses of the system, however takes a skilled person to do and takes time
- Sample forms
  - We can create forms to get information from current users, similar to surveys and questionaries.
- Sampling volume or work processes by system
- - Also known as studying documentation, we can simply take existing documents and work and write notes based off the information provided.
- look at documentation
- - read the system documentation and ensure it matches what the current system is doing
- Work shadowing
  - shadow a user in the current system to observe their interactions with the system

Quantity data collection	Qualitative data collection
Questionnaire	Interview
Survey	observation

6 Case Tools

- Computer aided software engineering tools help to design and implement applications in a project.
- They're often used to design the logical rather than the physical aspects of a project or solution
- Gantt and PERT charts are the most often used CASE tools in comp sci.

7 Gantt Charts

- A PERT Chart is also used to schedule, organise and coordinate activites within a prokect
- A PERT chart presents a graphic illustration of a project as a network diagram consisting of numbered nodes (circles) representing events, or milestones in the project linked by labelled vectors (directional lines) representing tasks in the project
- Direction of arrows of the lines indicated the sequence of tasks. These are called dependant or serial tasks.
- Tasks that must be completed in seuqnece but that dont require resources or completition time are considered to have event depenedncy. These are represented by dotted lines with arrows and are called dummy activities.
- May be preferred over Gantt chart because it clearly illustrates task dependencies. On the other hand PERT charts can be much more difficult to interpret, especially on complex projects.
- NEED TO PUT AN ARROW TO SHOW DEPENDINCYS!
- dont put arrows when tasks arent dependant
- A Gantt chart is a type of bar chart that illustrates a project schedule
- Modern day Gantt charts show the dependency between activiteis (eg. an activity that cant start until another has finished.
- Gantt charts provide an easy way to clearly see when a project phase can start based off the completed phases
- A gantt chart can be used for planning and scheduling of the project tasks. For example a gantt chart can be used to assess how long the project should take, determine the resouces needed, and plan the order in which tasks can be completed
- State the purpose of a Gantt chart
- To manage and keep track of a project

8 PERT Charts

- A program evaluation review technique chart is another CASE tool used to help in planning projects.
- Specifically they aid towards scheduling, organising and coordinating tasks within a project
- A pert chart shows the amount of time taken on each phase and in turn the critical, maximum and minimum pathways.
- Duration is drawn in hours/days/weeks in series with decencies
- A critical path is highlighted in red either red circle around project or as a red arrow.
- path the takes the longest time.
- START WITH TASK 0
- Describe how this diagram assists a manager in executing a project

- The PERT Chart can be used by manages to ensure that the project is accurately scoped. The manager has full view of the project before it is started and can then identify potential bottlenecks. They are best used for an overall view of the project dependencies and are not flexible enough to document small changes as the project evolves.

- which can be used to schedule, organise and coordinate tasks within a project. It will help the system analyst and their team of programmers, database developers and document writers to visualise the order of tasks, milestones and phrases within a project allowing the effective coordination or work across the project team.

9 Documentation

- Documentation is essential for any information system and/or product

- same documentation helps developers and designers to understand the ins and outs of the system or project as well as an integral part of planning

- Other documentation is created specifically to help the end user operate the system.

10 Context Diagrams

- A system context diagram, or context diagram (as its

- made up of entity's, data flows and the system

- The system is represented by a circle and has to have the word system in the name, to differentiate it from a process in a DFD

- Entitys are represented by a square/rectangle

- Data flows are curved lines with detail/information because it is representing data not physical objects. eg. Pen details

- context diagrams displays a systems relationships with external entity's, as such processes and data stores are not shown because they are part of the system.

- Mistakes
  - Black hole → data only going in, nothing coming out
  - Grey hole → different data coming out.
  - Miracle → data coming from database from nowhere

11 Data flow diagram L0 and L1

- processes are represented by circles, their labels are verbs → because they represent the system doing something. They must be numbered eg. 1.0, 2.0 this shows how deep the data flow diagram is.

- Internal data flows are data flows that not go to or from an entity.

- Balanced → data flows match that of the context diagram

- processes expected are 6-7

- Level 1 DFD → to deconstruct a complex system to understand it further

- recommended 3 processes in a level 1 DFD
- you don't need to show the entity, just the data flows
- external data flows must be balanced, inside can have extra data flows, data stores