

Working in teams

Introduction:

- ⊖ Software based systems will be huge, also software tool contains five million lines of code.
- ⊖ So that the work is shared between individual software developers within teams and between group of developers.
- ⊖ Team working will enhance the communication between individual developers and within teams and across teams.
- ⊖ Team - Group of people who are working together.
 - Small group environment
- ⊖ The term Project Team refers all the people working on a project.
- ⊖ The people who are working in project team may sit in different workgroups at some distance from each other.
- ⊖ These groups can also change over time.
- ⊖ Thus individual developers are transfer between teams during the period of project start and finish.
- ⊖ Team is created to do joint assignment
- ⊖ To perform the work assignments which are allocated to the staff, the organization needs one form of coordination between groups and individuals within a project.
- ⊖ Communication genres
 - refers Method of Communication
 - It is selected and developed to deal with particular need for project coordination.
 - The arrangements for communication between stakeholders are documented in communication plan
- ⊖ This Team work has an influence on all stages of step wise project planning framework.
 1. Identify Project scope and objectives
 2. Identify Project Infrastructure
 3. Analyze project characteristics

4. Estimate effort for each activity
5. Identify activity risks
6. Allocate resources
7. Review/publicize plan

Becoming a Team

- The organization first analyzes how the small work groups are formed.
- While forming a team it has five basic stages of development:
 - o Forming - Members of the group get to know each other
 - Try to set up some rules about behavior
 - o Storming - Conflicts arise to get leadership
 - Group's methods of operation are established
 - o Norming - Conflicts are largely settled
 - Group identity emerges
 - o Performing - Now tasks are at the hand
 - o Adjourning (Suspend or Stop)- Group disperse
- Some training activities such as management games are needed to promote team building and to people in the team work together.
- The team may consist of different types of people such as
 - o The chair - Good at running meetings
 - Strong but tolerant
 - o The plant - Good at generating ideas and solutions to the problems
 - o The monitor-evaluator - Good at evaluating ideas and solutions
 - Think well at selecting best one
 - o The shaper - Who helps to direct team's attention to the important issues
 - o The team Worker - Good at creating a good working environment
 - o The resource investigator - Skilled person to find resources ie) both physical

resources and information

- o The completer-finisher - Anxious (Worried) with completing tasks
 - o The company worker - Should be a good team player
 - Willing to take tasks for team success
- Problems occur when there is an imbalance between the role types of people in a group.

Group Performance`

- In many projects, some solutions are needed about which tasks are carried out collectively as a team and which are allotted to individuals.
- It is defined by “Some work yields better results if carried out as a team while some things are slowed down if the work is not partitioned on an individual basis”.
- The group tasks are categorized into:
 - o Additive Tasks - Effects of each participant are added to get final result
 - People involved are interchangeable
 - o Compensatory Tasks - Solutions of individual group members are pooled
 - Errors of some are compensated by the inputs from others
 - o Disjunctive Tasks - Means there is only one correct answer.
 - It depends on someone coming up with one right answer and others recognizing it as being correct
 - o Conjunctive Tasks - Means joining the tasks
 - Progress is governed by the rate of slowest performer
 - The overall task is not completed until all participants have completed.

Decision Making

- Decision can be categorized as
 - Structured
 - Simple
 - Routine
 - Straightforward rules
 - Unstructured
 - More complex
 - Requires degree of creativity

Mental Obstacles to good decision making

- Many decisions are made under pressure
- With incomplete information
 - Faulty Heuristics (Rule of thumb) – dangers are there
 - Information in hand-misleading
 - Stereotypes (well-known fact)
 - Escalation of commitment(difficult to alter once made a commitment)
 - Information Overload

Group Decision Making

- Decisions made by the team as a whole are more likely to be accepted than those that are imposed
- Complementary skills and expertise
- Communicate freely/get ideas
- Brainstorming techniques
- Aim is to have involvement of end users?
 - Prototyping and participatory approaches
 - JAD(Joint Application Development)

Barriers to good team decisions

- Inter-personal conflicts –team formation
- Conflicts tend to be dampened by emergence of *group norms* – shared group

opinions and attitudes

- *Risky shift* – people in groups are more likely to make risky decisions than they would as individuals

Delphi approach

To avoid dominant personalities intruding the following approach is adopted

1. Enlist co-operation of experts
2. Moderator presents experts with problem
3. Experts send in their recommendations to the moderator
4. Recommendations are collated and circulated to all experts
5. Experts comment on ideas of others and modify their own recommendation if so moved
6. If moderator detects a consensus, stop; else back to 4

Team ‘heedfulness’

- Football Team.
- Where group members are aware of the activities of other members that contribute to overall group success
- Impression of a ‘collective mind’
- Some attempts to promote this:
 - Egoless programming
 - Chief programmer teams
 - XP
 - Scrum

Egoless programming

- Gerry Weinberg noted a tendency for programmers to be protective of their code and to resist perceived criticisms by others of the code
- Encouraged programmers to read each others code
- Argued that software should become communal, not personal – hence ‘egoless programming’

Chief programmer teams

- Fred Brooks was concerned about the need to maintain ‘design consistency’ in large

softwaresystems

- Appointment of key programmers, Chief Programmers, with responsibilities for defining requirements, designing, writing and test software code
- Assisted by a support team: co-pilot – shared coding, editor who typed in new or changed code, program clerk who write and maintain documentation and tester
- Problem – finding staff capable of the chief programmer role

Extreme programming

XP can be seen as an attempt to improve team heedfulness and reduce the length of communication paths (the time between something being recorded and it being used)

- Software code enhanced to be self-documenting
- Software regularly refactored to clarify its structure
- Test cases/expected results created *before* coding – acts as a supplementary specification
- Pair programming – a development of the co-pilot concept

Scrum

- Named as an analogy to a rugby scrum – all pushing together
- Originally designed for new product development where ‘time-to-market’ is important
- ‘Sprints’ increments of typically one to four weeks
- Daily ‘scrums’ – daily stand-up meetings of about 15 minutes
- Unlike XP, requirements are frozen during a sprint
- At the beginning of the sprint there is a sprint planning meeting where requirements are prioritized
- At end of sprint, a review meeting where work is reviewed and requirements may be changed or added to.