2: Software Process Models.

Communication
Modeling
Construction
Deployment

Generic Process Model:

- Framework Activities: CPLMCD
- Umbrella Activities: Project tracking and control, risk management, quality assurance

Identify a task set: Task set defines the actual work to be done to accomplish the objectives of a software engineering action.

Process Flow:

Waterfall Process model: CPMCD
Pros: Easy plan and analysis and testing are straightforward.
Cons: Does not accommodate change well and testing occurs late in the process, customer approval at the end.

Prototyping Process Model: CPMCD - repeat the prototype until customer like it
Pros: Reduced impact of requirements changes & Customer involvement early and often & works well for small projects.
Cons: Customer involvement causes delays & hard to plan and manage.

Spiral Process Model: Improves on what has already been done.
Pros: Continuous customer involvement & good for large projects
Cons: Hard to manage, risk analysis failures can doom the project & need expert development team

Unified Process Model:
Pros: Quality documentation emphasized, continuous customer involvement, accommodates requirements changes.
Cons: Use cases are not always precise, requires expert development team.

Process Assessment and Improvement:

- The existence of a software process is no guarantee that software will be delivered on time or meet customer's needs.
- Any software process can be assessed to ensure that it meets a set of basic process criteria that have been shown to be essential for successful software engineering

Prescriptive Process Models: Advocate an orderly approach to software engineering.