program requirements and the design constraints

functional requirementswhat

nonfunctional requirementsthe

What a program Functional requirements needs to do. needs to do.

The manner Nonfunctional requirements in which the functional requirements need to in which the functional requirements need to be achieved. be achieved.

Functional Requirements

Input formats:

Nonfunctional Requirements

Performance requirements: Although

Real-time requirements:

choose

Design Constraints

User interface:

What the user sees, feels User interface and hears from the system. and hears from the system.

Platforms:

Typical and maximum input sizes:

A Few Pointers on Implementation

The most important rule is to be consistentespecially

Choose names carefully. In addition to being consistent in naming, try to make sure

names for functions and variables are descriptive.

Test before using a function or method.

Know thy standard library.

If possible, perform a review of your code.

1.5 Summary

Requirements Design Code implementation Unit testing Personal effort estimation User interface

Requirements

Size and Complexity

Size

Characteristics of Building a System

Technical Considerations of Development and Support

Problem and Design Decomposition

Technology and Tool Considerations

Database Network Middleware Other technical components such as code version control

Process and Methodology

A software development process is needed to guide and coordinate the group of people.

Effort Estimation and Schedule

Nontechnical Considerations of Development and Support

simple software project

team of one to three people,

people,

effort estimation and scheduling of the project is relatively easy.

functional and nonfunctional requirements

are fewer in number and complexity.

Assignments and Communications

The assignment of different people to different tasks such as testing, integration, or tool support requires more understanding of the skills of the people involved and the specific tasks they have to perform.

requires a deeper level of granularity and a finer level of scheduling.

Requirements of the Payroll System

Designing the Payroll System

Vertical and horizontal design entities.

horizontal entities are the com- mon service functions such as the error handler that crosses all the individual application features.

vertical entities are the different application domain-specific functions such as the tax and benefits deduction function in a payroll system.

Code and Unit Testing the Payroll System

Integration and Functionally Testing the Payroll System

Release of the Payroll System

formally collected from the individual programmers.

collection activity is known as integration,

modules

Support and Maintenance

2.4 Summary

growth in the breadth and depth of complex software problems and the associated software solutions.

technical and non- technical issuesfrom

design decomposition and communications to process, tools, and

methodologywere

Large software projects require the control of the process, the product, and the people involved.

field of software engineering is vital

to

development and support of these large and complex systems.

domain model