

## Buzzword Cards

---

Note: Cards are organized by the topic under which were discussed and are further subdivided by the source from which they were obtained. Sources can be "Book" which refers to Pressman's text *Software Engineering: A Practitioner's Approach 8th Edition* or "Notes" which references course notes from CS690 as taught by Dr. Wayne Zage.

### Managing Software Projects - Book:

- 1) {'Resources': ['Human', 'Software', 'Environmental', 'Hardware']} : pg. 732
- 2) {'Decomposition': ['Break', 'Parts', 'Smaller', 'Problem']} : pg. 734

### User Interface Design - Book:

- 3) {'Analysis': ['Design', 'Implementation', 'Document', 'Critical']} : pg. 317

### Process Models - Book:

- 4) {'Prescriptive Model': ['Structure', 'Order', 'Sequential', 'Process']} : pg. 41
- 5) {'Process Model': ['Activity', 'Action', 'Flow', 'Software']} : pg. 41
- 6) {'Prototyping': ['Software', 'Requirement', 'Model', 'Evolutionary']} : pg. 45

### Risk Management - Book:

- 7) {'Performance Risk': ['Requirements', 'Uncertainty', 'Cost', 'Support']} : pg. 782
- 8) {'Cost Risk': ['Budget', 'Maintain', 'Price', 'Uncertainty']} : pg. 782
- 9) {'Support Risk': ['Uncertainty', 'Correct', 'Help', 'Enhance']} : pg. 782
- 10) {'Schedule Risk': ['Uncertainty', 'Project', 'Time', 'Deliver']} : pg. 782
- 11) {'Risk Estimation': ['Projection', 'Likelihood', 'Impact', 'Determine']} : pg. 782
- 12) {'RMMM': ['Plan', 'Probability', 'Strategy', 'Minimize']} : pg. 785
- 13) {'Cutoff Line': ['Risk', 'Terminate', 'Estimate', 'Identification']} : pg. 784
- 14) {'Fire-fighting': ['Crisis', 'Panic', 'Project', 'Problem']} : pg. 778
- 15) {'Uncertainty': ['Unknown', 'Unsure', 'Minimize', 'Risk']} : pg. 778

### Product Metrics - Book:

- 16) {'External Inputs': ['User', 'Outside', 'Function', 'Output']} : pg. 659
- 17) {'External Outputs': ['User', 'Information', 'Report', 'Input']} : pg. 659
- 18) {'Measure': ['Size', 'Dimension', 'Extent', 'Capacity']} : pg. 655

### Software Engineering - Book:

19) {'Mongolian Horde': ['Add', 'Behind', 'Mythical Man Month', 'Schedule', 'Productivity']} : pg. 24

### Modeling - Book:

20) {'Requirements Engineering': ['Communication', 'Needs', 'Process', 'Project']} : pg. 132

21) {'Inception': ['Start', 'Beginning', 'Product', 'Idea']} : pg. 133

22) {'Ubiquitous Computing': ['Processors', 'Distributed', 'Everywhere', 'Commonplace']} : pg. 133

23) {'Elicitation': ['Ask', 'Objective', 'Accomplish', 'Use']} : pg. 134

24) {'Problem of Scope': ['Boundary', 'Space', 'Domain', 'Context']} : pg. 134

25) {'Problem of Understanding': ['Unsure', 'Limitations', 'Communication', 'Partial']} : pg. 134

26) {'Problem of Volatility': ['Requirements', 'Change', 'Time', 'Quickly']} : pg. 134

27) {'Goal-Oriented': ['Long-Term', 'Purpose', 'Achieve', 'Requirements']} : pg. 134

28) {'Elaboration': ['Expand', 'Elicitation', 'Inception', 'Refined']} : pg. 134

29) {'User Scenario': ['Describe', 'Actors', 'Scene', 'Workflow']} : pg. 135

30) {'Negotiation': ['Compromise', 'Reconcile', 'Argue', 'Requirements']} : pg. 135

31) {'Iterative Approach': ['Sequential', 'Technique', 'Systematic', 'Requirements']} : pg. 135

32) {'Specification': ['Document', 'Model', 'Prototype', 'Description']} : pg. 135

33) {'Validation': ['Verify', 'Assess', 'Quality', 'Review']} : pg. 136

34) {'Requirements Managment': ['Activities', 'Life Cycle', 'System', 'Track']} : pg. 138

35) {'Software Engineer': ['Design', 'Manage', 'System', 'Programmer']} : pg. 138

36) {'Collaboration': ['Together', 'Group', 'Coordinate', 'Work']} : pg. 140

37) {'Questions': ['Ask', 'Gather', 'Query', 'Inquire']} : pg. 140

38) {'Business Need': ['Identify', 'Recognize', 'Project', 'Inception']} : pg. 133

39) {'Task Analysis': ['Perform', 'Circumstance', 'Sequence', 'Workflow']} : pg. 326

40) {'Task Elaboration': ['Function', 'List', 'Breakdown', 'Design']} : pg. 328

41) {'Workflow Analysis': ['Multiple', 'Process', 'User', 'Technique']} : pg. 328

42) {'Swimlane': ['Diagram', 'Pool', 'Water', 'Users']} : pg. 329

43) {'Design Pattern': ['Abstraction', 'Solution', 'Interface', 'Common']} : pg. 334

44) {'Response Time': ['Delay', 'Lag', 'Bad', 'Variable']} : pg. 335

45) {'Accessibility': ['Impaired', 'Disability', 'Usable', 'Guidelines']} : pg. 336

### The Nature Of Software - Book:

46) {'Embedded Software': ['Firmware', 'Hardware', 'Controller', 'Processor', 'PROM']} : pg. 7

47) {'Cloud Computing': ['Internet', 'Remote', 'Resource', 'Azure/Google/AWS']} : pg. 10

48) {'Process Pattern': ['Solution', 'Resolution', 'Template', 'Planning']} : pg. 35

49) {'Modeling': ['Sketch', 'Mock-up', 'Miniature', 'Scale']} : pg. 17

50) {'Customer': ['Request', 'Pay', 'Requirements', 'Person']} : pg. 17

51) {'Obsolete': ['Outdated', 'Expired', 'Extinct', 'Deprecated']} : pg. 22

52) {'HTML': ['Hypertext', 'HTTP', 'Protocol', 'Browser']} : pg. 10

53) {'Algorithm': ['Code', 'Program', 'Processor', 'Computer']} : pg. 21

54) {'Compiler': ['Assemble', 'Execute', 'Program', 'Code']} : pg. 7

Understanding Requirements - Book:

55) {'requirement': ['Customer', 'Model', 'Change', 'Design']} : pg. 131

Software Testing Strategies - Book:

56) {'Independent Test Group': ['Separate', 'Complete', 'Procedure', 'Correct']} : pg. 469

Software Quality Assurance - Book:

57) {'Standards': ['Ensure', 'Conform', 'Voluntary', 'SQA']} : pg. 450

58) {'Audit': ['Review', 'Conformity', 'Quality', 'Certification']} : pg. 450

59) {'Six Sigma': ['Standard', 'High', 'Define', 'Measure']} : pg. 458

60) {'Failures-in-time': ['Statistical', 'Measure', 'Reliability', 'Billion']} : pg. 460

61) {'Mean-time-to-failure': ['Average', 'Duration', 'Program', 'Crash']} : pg. 450

Software Process Structure - Book:

62) {'Software Engineering': ['Process Management', 'System', 'Development', 'Discipline']} : pg. 15

63) {'Process Flow': ['Iterative', 'Linear', 'Parallel', 'Umbrella']} : pg. 31

64) {'Umbrella Activities': ['tracking', 'risk', 'quality', 'configuration', 'technical']} : pg. 31

65) {'Communication': ['Talk', 'Speak', 'Telephone', 'Internet']} : pg. 34

Process and Project Metrics - Book:

66) {'Defect': ['Error', 'Bug', 'Crash', 'Failure']} : pg. 450

Quality Management - Book:

67) {'Quality of Conformance': ['Degree', 'Requirements', 'End-user', 'Comply']} : pg. 414

Project Management Concepts - Book:

68) {'Stakeholders': ['Managers', 'Practitioners', 'Customers', 'End User']} : pg. 687

69) {'Organizational Paradigms': ['Closed', 'Random', 'Open', 'Synchronous']} : pg. 690

70) {'Agile': ['Satisfaction', 'Incremental', 'Informal', 'Simplicity']} : pg. 691

71) {'Software Scope': ['Context', 'Information', 'Function', 'Features']} : pg. 694

Quality Management - Book:

72) {'User Interface': ['Input', 'Output', 'Keyboard', 'Screen', 'Website']} : pg. 672

73) {'Cyclomatic Complexity': ['Linear', 'Independent', 'Metric', 'Testing']} : pg. 455

74) {'Validation Testing': ['Requirements', 'Software', 'Verify', 'Develop']} : pg. 484

75) {'System Testing': ['Mesh', 'Integrate', 'High', 'Performance']} : pg. 471

76) {'Smoke Testing': ['Integration', 'Development', 'Time-critical', 'Assessment']} : pg. 479

77) {'Alpha Tests': ['Developer', 'End-User', 'Process', 'Integrated']} : pg. 485

78) {'Beta Test': ['End-user', 'Developer', 'Live', 'Environment']} : pg. 485

79) {'Stress Testing': ['Load', 'Volume', 'Resources', 'Demand']} : pg. 487

- 80) {'Configuration Testing': ['Environment', 'Deployment', 'Documentation', 'Installed']} : pg 487  
81) {'Security Testing': ['Penetration', 'Vulnerability', 'Attack', 'Hack']} : pg. 486  
82) {'Recovery Testing': ['Failure', 'Automatic', 'Human-intervention', 'Resilience']} : pg. 486

#### Design Concepts - Book:

- 83) {'Pseudocode': ['Design', 'Program', 'Flow', 'Language']} : pg. 247

#### Use Case-based Requirements - Notes:

- 84) {'Use case': ['Object Oriented', 'Actors', 'Actions', 'Oval']} : pg. 2  
85) {'Scenario': ['Use case', 'Actions', 'Sequence', 'Behavior']} : pg. 2  
86) {'Actor': ['User', 'Use Case', 'Information', 'Stick']} : pg. 3  
87) {'Use Case Diagram': ['Use Case', 'Actors', 'Visual', 'Picture']} : pg. 5  
88) {'Preconditions': ['Main Flow', 'Use Case', 'Before', 'Start']} : pg. 6  
89) {'Main Flow': ['Use Case', 'Actor', 'Steps', 'Sub flow']} : pg. 6  
90) {'extend relationship': ['condition', 'optional behavior', 'scenario', 'exceptional']} : pg. 9

#### Object-Oriented Analysis and Design - Notes:

- 91) {'OO Analysis': ['investigation', 'describe', 'concept', 'problem domain']} : pg. 1  
92) {'OO Design': ['define ', 'logical software objects', 'implement', 'Object']} : pg. 1  
93) {'Conceptual model': ['abstract', 'object', 'block', 'attributes']} : pg. 2  
94) {'collaboration diagram': ['message', 'data flow', 'illustrating', 'function']} : pg. 2  
95) {'class diagrams': ['components', 'object block', 'combined model', 'function and attribute']} :  
pg. 3  
96) {'UML': ['object-oriented analysis', 'methodology', 'visualizing', 'development life cycle']} : pg.  
3  
97) {'multiplicity': ['relationship', 'interaction', 'objects', 'number']} : pg. 8  
98) {'navigation': ['direction', 'multiplicity', 'arrow', 'association']} : pg. 8  
99) {'inheritance': ['relationship', 'super-class', 'sub-class', 'common attributes']} : pg. 8  
100) {'include relationship': ['function', 'reuse', 'use case', 'consequence']} : pg. 8  
101) {'misuse case': ['privacy and security', 'hostile', 'hacker', 'threat']} : pg. 10  
102) {'actors': ['interact', 'system', 'player', 'relationship']} : pg. 5  
103) {'attributes': ['class', 'definition', 'requirement', 'variable']} : pg. 7  
104) {'operations': ['function', 'interaction', 'behavior', 'parameters']} : pg. 7  
105) {'class': ['Types', 'Primitives', 'Inheritance', 'School']} : pg. 4

#### Risk Management - Notes:

- 106) {'Risk Strategy': ['Reactive', 'Proactive', 'Plan', 'Minimize']} : pg. 1  
107) {'Software Risks': ['Project', 'Technical', 'Business', 'Categories']} : pg. 2  
108) {'Risk Identification': ['Size', 'Impact', 'Probability', 'Quantify']} : pg. 2

#### What is project planning - Notes:

- 109) {'Planning': ['Task', 'Organization', 'Design', 'Dependency']} : pg. 1  
110) {'Metrics': ['Models', 'Predicts', 'Measures', 'System']} : pg. 2

- 111) {'Critical Path': ['Plan', 'Complete', 'Longest', 'Project']} : pg. 2
- 112) {'Function Points': ['Measure', 'Compute', 'LOC', 'Cost']} : pg. 4
- 113) {'Empirical Validation': ['Analyze', 'Data Collection', 'Proof', 'Experiment ']} : pg. 2
- 114) {'Effort': ['Cost', 'Duration', 'Work time', 'Measure']} : pg. 2
- 115) {'Risk': ['Avoid', 'Estimate', 'Evaluate', 'Analyze']} : pg. 1
- 116) {'Estimates': ['Approximation', 'Calculation', 'Guess', 'Analyze']} : pg. 3
- 117) {'Size Metrics': ['Operators', 'Operands', 'Occurrences', 'LOC']} : pg. 3
- 118) {'Macro-Model': ['Scale', 'Productivity', 'Big', 'Size']} : pg. 4
- 119) {'Early Planning': ['Estimate', 'Schedule', 'Tasks', 'Dependency']} : pg. 1
- 120) {'Emperical': ['Verified', 'Science', 'Theoretical', 'Data']} : pg. 2
- 121) {'Code Metric': ['Code', 'Data structure', 'Complexity ', 'Logic']} : pg. 3
- 122) {'Effort estimation': ['Past experience ', 'Data', 'Formal model', 'Average methods']} : pg. 4
- 123) {'Total effort': ['Development cost', 'Theoretical', 'Statistical', 'Mathematical']} : pg. 4
- 124) {'Criteria': ['Validity', 'Judge', 'Model', 'Objectivity']} : pg. 5
- 125) {'COCOMO': ['Tool', 'Estimate', 'Schedule', 'Cost']} : pg. 5
- 126) {'Product Attributes': ['Software Reliability', 'Data base', 'Complexity ', 'Cost driver']} : pg. 6
- 127) {'Personnel Attributes': ['Analyst Capability', 'Application Experience', 'Programmer Capability', 'Programming Language ']} : pg. 6
- 128) {'Computer Attributes': ['Execution Time', 'Storage Constraint', 'Turnaround time', 'Virtual Machine']} : pg. 6
- 129) {'Estimating Techniques': ['Algorithmic', 'Top-Down', 'Bottom-Up', 'Expert Judgment', 'Analogy']} : pg. 8

#### Software Quality Assurance - Notes:

- 130) {'Reliability': ['Crash', 'Intended', 'Test', 'Quality']} : pg. 1
- 131) {'Process Specific Activities': ['KPA', 'Maturity', 'Levels', 'Testing']} : pg. 3
- 132) {'Quality Assurance': ['Review', 'Methods', 'Procedures', 'Guarateee']} : pg. 3
- 133) {'Prevention Costs': ['Quality', 'Planning', 'Reviews', 'Training']} : pg. 2
- 134) {'Failure Cost': ['Analysis', 'Project', 'Retest', 'Postmortem']} : pg. 2

#### Introduction to software engineer: - Notes:

- 135) {'Confidentiality': ['Employer', 'Secret', 'Respect ', 'Leak']} : pg. 8

#### What is Project Planning - Notes:

- 136) {'BRAC': ['Code', 'Requirements', 'Estimation', 'Volatility']} : pg. 7

#### Design Techniques - Notes:

- 137) {'Data Flow Oriented Design': ['process', 'DFD', 'structure', 'program']} : pg. 1
- 138) {'Top-Down Function Design': ['function', 'module', 'software', 'information driven']} : pg. 1

#### Notes on software testing - Notes:

- 139) {'Path Testing': ['Input', 'Output', 'Execution', 'Follow']} : pg. 11
- 140) {'Symbolic Evaluation': ['Methods', 'Dynamic', 'Global', 'Testing']} : pg. 10

#### The Structured Specification - Notes:

- 141) {'Data Dictionary': ['Document', 'Meaning', 'Description', 'Terms']} : pg. 3
- 142) {'DFD': ['Design', 'Data', 'Diagram', 'Flow']} : pg. 1
- 143) {'Functional Specification': ['Analyst', 'Requirements', 'User', 'Document']} : pg. 1
- 144) {'Data Store': ['DFD', 'Keep', 'Diagram', 'Design']} : pg. 1
- 145) {'Terminators': ['DFD', 'External', 'Sources', 'Sinks']} : pg. 1
- 146) {'Contextual Level': ['DFD', 'Zero', 'Top', 'Design']} : pg. 2
- 147) {'Decision Trees': ['Design', 'Diagram', 'Branch', 'Picture']} : pg. 4
- 148) {'ERDs': ['Entity', 'Diagram', 'Design', 'Data']} : pg. 4
- 149) {'Functional Primitive': ['DFD', 'Last', 'Design', 'Break']} : pg. 4
- 150) {'Structured Specification': ['functional specification', 'readings', 'user', 'document']} : pg. 1
- 151) {'Terminator': ['DFD', 'external data sources', 'diagrams', 'destinations']} : pg. 1
- 152) {'Component Variations': ['flow type', 'process', 'store', 'operations']} : pg. 2

#### Design Principles - Notes:

- 153) {'Abstraction': ['Module', 'Object', 'Class', 'Design']} : pg. 1
- 154) {'Software Design': ['Computer Science', 'Structure', 'System', 'Analysis']} : pg. 1
- 155) {'Fan-out': ['Module', 'Fan-in', 'Call', 'Number']} : pg. 3
- 156) {'Fan-in': ['Module', 'Fan-out', 'Number', 'Call']} : pg. 3
- 157) {'Cohesion': ['Module', 'Association', 'Function', 'Coupling']} : pg. 3
- 158) {'Coupling': ['Module', 'Class', 'Interdependence', 'Cohesion']} : pg. 3
- 159) {'Functional Cohesion': ['Module', 'All', 'Task', 'Function']} : pg. 4
- 160) {'Sequential Cohesion': ['Module', 'Input', 'Output', 'Function']} : pg. 4
- 161) {'Communication Cohesion': ['Input', 'Output', 'Same', 'Function']} : pg. 4
- 162) {'Procedural Cohesion': ['Different', 'Control', 'Function', 'Unrelated']} : pg. 4
- 163) {'Temporal Cohesion': ['Function', 'Time', 'Module', 'Permanent']} : pg. 4
- 164) {'Logical Cohesion': ['Function', 'Outside', 'Flag', 'Execute']} : pg. 4
- 165) {'Coincidental Cohesion': ['Function', 'Random', 'Accidental', 'Unexpected']} : pg. 4

#### Introduction to software engineer - Notes:

- 166) {'Reuse Model': ['Refine', 'Object', 'Copy', 'Extract']} : pg. 5

#### What is project planning: - Notes:

- 167) {'Code Modification': ['Change', 'Reuse', 'Modify software', 'Assessment']} : pg. 6

#### Introduction to Software Engineering - Notes:

- 168) {'Waterfall Model': ['Feedback Loops', 'Documentation-driven', 'Easier Maintenance', 'Lacking Client Feedback', 'Tiered']} : pg. 2

#### Notes on Software Testing - Notes:

- 169) {'Black Box Testing': ['Specifications', 'API', 'Inputs', 'Outputs']} : pg. ?
- 170) {'White Box Testing': ['Method', 'Known', 'Source', 'Inside']} : pg. ?
- 171) {'Naive Testing': ['Simplistic', 'Execution', 'Cases', 'Guessing']} : pg. ?

- 172) {'Exhaustive Path': ['Brute', 'All', 'Combinations', 'Traverse']} : pg. ?
- 173) {'Test Plan': ['Management', 'Coverage', 'Verify', 'Box']} : pg. 3
- 174) {'Random Input Testing': ['Bad', 'Poor', 'Stochastic', 'Noise']} : pg. 4
- 175) {'Equivalence Partition': ['Sets', 'Intersection', 'Union', 'Divide']} : pg. 4
- 176) {'Boundary Value Analysis': ['One', 'Limits', 'Ranges', 'Condition']} : pg. 4
- 177) {'Error Guessing': ['Experience', 'Analysis', 'Intuition', 'Knowledge']} : pg. 4
- 178) {'Logic Coverage': ['Statement', 'Path', 'Branch', 'Condition']} : pg. 11
- 179) {'Infeasible Path': ['Unreachable', 'Traversal', 'Execution', 'Impossible']} : pg. 12
- 180) {'Statement Coverage': ['Verify', 'Once', 'Logic', 'Strategy']} : pg. 12
- 181) {'Path Coverage': ['Potential', 'Conditions', 'All', 'Combinations']} : pg. 13
- 182) {'Condition Coverage': ['Test', 'Each', 'Logical', 'Boolean']} : pg. 14
- 183) {'Defect Density Model': ['Testing', 'Time', 'Effort', 'Minimized']} : pg. 18
- 184) {'Saturation Effect': ['False', 'Reliability', 'Estimates', 'Misled']} : pg. 18

#### Introduction to software engineering - Notes:

- 185) {'Extreme Programming': ['Stories', 'Features', 'Builds', 'Test Cases', 'Pair', 'Continuous']} : pg. 3
- 186) {'Synchronize and Stabilize': ['Microsoft', 'Requirements', 'Specifications', 'End of day']} : pg. 4
- 187) {'Fountain Model': ['Waterfall', 'Cycles', 'Serial', 'Dependency']} : pg. 4
- 188) {'Spiral Model': ['Risk', 'Precede', 'Evaluate', 'Plan']} : pg. 4
- 189) {'Object Oriented Programming': ['Class', 'Object', 'Inheritance', 'Encapsulation']} : pg. 4
- 190) {'Unit test': ['Program', 'Part', 'Code', 'Function']} : pg. 2
- 191) {'Maintenance': ['Cost', 'Time', 'During', 'Bug']} : pg. 2
- 192) {'Documentation': ['Text', 'Customer', 'Tutorial', 'Guideline']} : pg. 6
- 193) {'Maintainability': ['Needs', 'Software', 'Technical support', 'Time']} : pg. 7
- 194) {'Dependability': ['Reliable', 'Robust', 'Availability', 'System']} : pg. 7
- 195) {'Efficiency': ['Trustworthy', 'Unused resources', 'Efficient', 'Useful output']} : pg. 7
- 196) {'Usability': ['Discover-able', 'Usable', 'Satisfaction', 'Errors']} : pg. 7

#### Evaluating Software Models - Notes:

- 197) {'relative error': ['actual development effort', 'estimated development effort', 'software model', 'predictive value']} : pg. 1
- 198) {'mean relative error': ['mean', 'sum', 'predict', 'criteria']} : pg. 1
- 199) {'magnitude of the relative error': ['bias', 'absolute', 'adjust', 'magnitude']} : pg. 1
- 200) {'Mean magnitude of relative error': ['mean', 'effort estimation', 'average', 'acceptable']} : pg. 1