


	NOVICE	INTERMEDIATE	ADVANCED
 <p>DESIGN</p>	<ul style="list-style-type: none"> ✓ No (or minimal) branching ✓ Configuration as code ✓ API management ✓ Library management 	<ul style="list-style-type: none"> ✓ Push business metrics ✓ Maintain product backlog (technical debt) ✓ Practice release planning 	<ul style="list-style-type: none"> ✓ Documented coding standards ✓ Document organizational design patterns ✓ Practice "spike" solutions
 <p>DEVELOP</p>	<ul style="list-style-type: none"> ✓ Versioned code base ✓ Auto triggered build (commit hooks) ✓ Scripted builds ✓ Basic scheduled builds ✓ Dedicated build server ✓ Scripted configuration changes ✓ Standard process for all environments ✓ Stored builds ✓ Feature branching 	<ul style="list-style-type: none"> ✓ Automated tag & versioning ✓ Build once, deploy anywhere ✓ Development team sends pull request to leads for architecture reviews ✓ Tool based code reviews ✓ Test driven development 	<ul style="list-style-type: none"> ✓ Zero downtime deploys ✓ Multiple build machines ✓ Strong focus on feedback loops via continuous integration processes ✓ Notifications and pro-active monitoring from continuous integration process
 <p>TEST</p>	<ul style="list-style-type: none"> ✓ Automatic unit tests ✓ Automatic integration tests 	<ul style="list-style-type: none"> ✓ Automatic components tests (isolated) ✓ Separate test environment ✓ Some automatic acceptance tests 	<ul style="list-style-type: none"> ✓ Full automatic acceptance tests ✓ Verify expected business value ✓ Automatic performance tests ✓ Automatic security tests
 <p>RELEASE</p>	<ul style="list-style-type: none"> ✓ Documented manual deploy ✓ Some deployment scripts exist ✓ Push-button deployments to test environments ✓ First step towards standardized deploys 	<ul style="list-style-type: none"> ✓ Basic pipeline with deploy to production ✓ Automated deployments after tests pass ✓ Auto deploy to first test environment ✓ Standard deployments across all environments ✓ Push-button deployments to production 	<ul style="list-style-type: none"> ✓ Ability to implement continuous deployment (zero touch) ✓ Full automatic database deployment ✓ Multi-tier deployments



INFORMATION & REPORTING

- Scheduled quality reports
- Baseline process metrics
- Common information model
- Fully Traceable
- Static code analysis

- Change request to production deployment
- Automated release notes
- Dynamic graphing and dashboards
- Dynamic test coverage analysis
- Report trend analysis



CULTURE & ORGANIZATION

- Prioritize work
- One team backlog
- Define and document process
- Frequent commits
- Cross functional team collaboration
- Component ownership
- Continuous improvement
- Common change process
- Tool based code review process
- Team responsible all the way to production