

1. Place the green testability post-it notes into the categories, you believe they belong to.
2. In a different colour to green/yellow, add your own questions to each testability category.
3. Place the yellow automatability post-it notes into the categories, you believe they belong to.
4. In a different colour to green/yellow (and your chosen testability), add your own question to each automatability category.

The 10 P's of Testability



We thing steps for CI can be placed in a couple of p's

	People	Philosophy	Product	Process	Project	Problem	Pipeline	Productivity	Production issues	Proactivity
Testability	<p>Who has all the product knowledge?</p> <p>Are the developers encouraged to do exploratory testing?</p> <p>What testing skills do the team have?</p> <p>Do you expect to grow the team?</p>	<p>Does the whole team take responsibility for testing?</p> <p>What kind of development process (TDD, BDD etc.)?</p>	<p>How easy is it to create test data?</p>	<p>Are stories adequately broken down?</p> <p>Who is involved in story planning?</p>	<p>Does the project have a deadline?</p> <p>How many test environments are there?</p> <p>Does the project have a deadline?</p>	<p>What risks are currently the most important?</p> <p>Do you know who your customers are?</p>	<p>What are the steps for the currently CI system?</p> <p>How long does a build currently take?</p>	<p>What is the mean time to discovery for high impact bugs?</p> <p>How much testing is done as early as possible?</p>	<p>How many known bugs are currently in production?</p> <p>How do customers report issues in production?</p>	<p>Is the team actively encourage to experiment?</p> <p>How often does the team have retrospectives?</p>
Automatability	<p>What automation tools does the team have experience with?</p>	<p>Do the developers contribute to the quality of the automated check code?</p> <p>Does the whole team contribute to automated checks?</p>	<p>How many APIs does the product have?</p> <p>Does the UI have identifiers to make it easier to automate?</p>	<p>Is there a big backlog of things waiting to be automated?</p> <p>Does the automation efforts keep up each sprint?</p>	<p>Does the project have budget for test environments?</p> <p>Who is going to be maintaining the automated checks?</p> <p>Has the project estimated for time to set up an automation framework?</p> <p>Does the project have budget for test environments?</p>	<p>Do we know what flows the customers value the most?</p> <p>Do we understand what risks need to be automated?</p>	<p>When is the product deployed to test/production etc?</p> <p>Does the existing pipeline contain any automated checks?</p> <p>Is there a pipeline already in place?</p>	<p>Are the automated checks ran as early as possible to provide rapid feedback?</p> <p>Are problems with the automation framework identified early?</p>	<p>Are there any known issues that would impact our automation efforts?</p> <p>Is there any monitoring in place for identifying potential production issues?</p>	<p>Is the team open to new tooling?</p> <p>Does the team have regularly retrospectives on the automation efforts?</p>

### Product questions

- Give us a demo :)
- What platforms are we using?
- How does the network/server landscape looks like (loadbalancer/machines)?
- What is the architecture (connections to other systems)?
- What kind of data transfer technologies are being used (API, etc)?
- Are there background processes?
- What tests are being executed at this moment?
- Features Parts Protocols Layers Technology Users
- Should it be secured (HTTPS)?
- Are the features dependant on each other?
- What features does the product have?
- What is the expected uptime of the app?
- What kind of users does the product have?
- What are the key user stories?
- How many users should the product service at once?
- Which language is it written in?
- How should the UI look?
- What is the expected performance?
- How many users does the product have
- What problem is the product solving
- What frameworks are being used?
- Do we need to have authentication \ authorization?
- Do we need to use any 3rd party services?
- Do we need any reporting \ monitoring mechanism?
- what are the accessibility requirements?
- What needs to change to deliver the software to a new client?
- Data storage (sql/nosql/flat files)
- code architecture?
- supported languages? (localization)
- What are the main user flows?
- What is the roadmap for the product?
- Is it a monolithic /microservices?
- What is being logged?