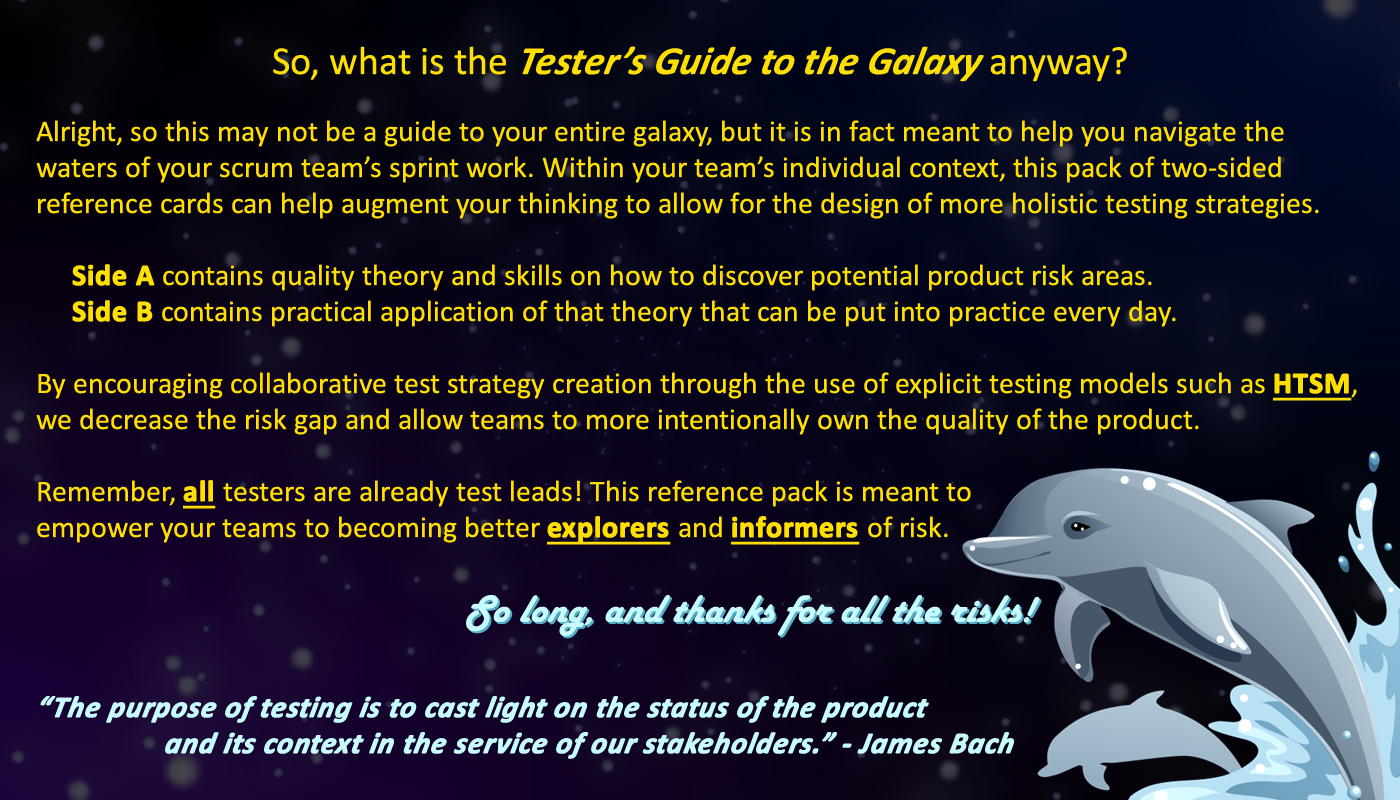
|  |  |
| --- | --- |
| Building A Test Strategy – Layer 3: Hardening At this point, the team has gained much more knowledge than they had in Sprint Planning, so it’s time to gain team approval on testing strategy and begin shifting more toward our primary testing activities. Key takeaways:  * Leverage the Wisdom of the Crowd   + The collective opinion of the group informs better than that of a single individual. * Poll the Team & Expose Assumptions:   + Targeted discussions with Testers, Developers, Product Owners, and Scrum Masters.   **(continued on back) Card 6A** | Backlog Grooming The team and perhaps external subject-matter experts may be needed for this process. Team Members can begin using explicit models to help the team better grasp development and testing effort. Key takeaways:  * Models as Litmus Tests   + From *HTSM* use the ***Quality Criteria*** and ***Product Elements*** node as a barometer to aid in story acceptance.   **(continued on back) Card 2A** |
| Dev Complete By this point, test strategies are vetted and have team ownership. Testing continues according to the agreed upon plan. Pair-testing is practiced when appropriate. Key takeaways:  * Guided Testing   + Hardened test strategies aid in rapid experimentation that provide high-value by exposing risk within the product. * Execute Pair Testing   + Perform ***Shake And Bake*** process on previously decided candidates, upon initial code completion, typically pre-commit.   **(continued on back) Card 7A** | Sprint Planning Each scrum team can better support our business goals by ensuring that we’re more informed when reevaluating story estimates. We do this by using guideword heuristics, as well as mnemonics from the larger community to help us make decisions. Key takeaways:  * Use Guideword Heuristics (rules of thumb for making decisions)   + From HTSM use the ***Quality Criteria*** and ***Test Techniques*** nodes to make more educated judgements. * Use Mnemonics (a character pattern that aids in recall). * Decide which stories are good ***Shake And Bake*** candidates.   **(continued on back) Card 3A** |
| Pre-Release Releases cadences can take all shapes and sizes: daily, weekly, multi-item, off-hours, etc; however, scrum teams can use these guard-rails to help become better informed within their own context. Key takeaways:  * Product Confidence   + Teams can build release-consistency by reviewing their own ***Release Checklist*** that acts as a template for smoke and sanity checks related to your team’s product area. * Reversion Hypotheticals   + High-risk stories may benefit from having pre-defined rollback processes.   **(continued on back) Card 8A** | Building A Test Strategy – Layer 1: Creation The first day of the sprint is usually filled with a variety of conversations that prep the team for the work ahead. Get a jumpstart on this by establishing testing strategies and getting buy-in early in the sprint. Key takeaways:  * Test Strategies for Each Story   + Collaborate with team members using the ***Risk Analysis Heuristics (for Digital Products)*** and HTSM nodes, ***Quality Criteria*** and ***Test Techniques****.* * Heavy Learning   + Good test strategies begin with a lot of unanswered questions. Seek out the necessary subject-matter experts to break these down.   **(continued on back) Card 4A** |
| Release & Follow Through At this point the team’s confidence level in the product developed should be high, such that your release process becomes only a formality. Use these reminders to fully follow through and round out your sprint. Key takeaways:  * Lean Production Testing   + Environment stability allows teams to require only sanity checks and smoke tests during most releases. * Inform (Yes) vs. Gatekeeping (No) * Compelling Story: Three-Part Model * Post-Deployment Monitoring * Team Retro: Remaining risk gap, action items, re-poll the team.   **(continued on back) Card 9A** | Building A Test Strategy – Layer 2: Iteration By now you have enough information to have made a good dive into test strategy creation for each story. Continue gathering information. Don’t wait until you finish an outline/mind-map to share it, be in continual communication with your other team members. Key takeaways:  * Continue Test Strategy Creation   + Have the team start evaluating your test strategies. Make modifications based on feedback from product management. * Constant Transparency   + Test strategies are living documents that should be continually modified and made available to the team.   **(continued on back) Card 5A** |

|  |  |
| --- | --- |
| *gauge.png*Put It Into Action! ***Models As Litmus Tests for Story Acceptance***   1. For grooming sessions, provide printouts of the   ***Quality Criteria*** & ***Product Elements*** from HTSM.   1. Talk through the child nodes from HTSM to help the group become more aware of product risks, as well as potential process roadblocks that could augment the development process in ways not yet anticipated.   ***Why do this?***  *A minimal code change in one area of the product may have minor implications for your team’s area of focus. However, after discussing the* ***Data*** *and* ***Platform*** *nodes, your team exposes a dependent feature that resides in another Release Train’s realm. This may require Product Management to take a more deliberate and coordinated approach and could likely modify the delivery cadence.*  **(continued on back) Card 2B** | group.pngIntellectual Humility Seeks Counsel ***Exposing Assumptions, Uncovering Unknowns***  Once your product outlines have become hardened into  team-approved test strategies, you can move on to some of the more practical testing activities, per what is appropriate to your product and team’s context.   1. Put your testing artifact into the format that provides the most value for your team.    1. For example, some testing may benefit from the granularity of scripted test cases, while a mind-map format would be sufficient for another. 2. **Poll The Team:** Testers, ask the following…    1. Developers: *What roadblocks are you experiencing? What new information have you discovered since our planning session?*    2. Product Owners: *Have there been any shifts in the customer's thinking that might affect current sprint items?*    3. Scrum Masters: *Is there anything I am doing that might be causing friction?*   **(continued on back) Card 6B** |
| *map.png*Follow Your Nose! ***Guideword Heuristics to Decrease the Risk Gap***   1. In Sprint Planning, provide printouts of the ***Quality Criteria***,   ***Product Elements*** and ***Test Techniques*** from HTSM.   1. Use the child nodes from HTSM to fuel more informed discussions and fill gaps in the team’s thinking. Treat this as a guide rather than a checklist, since some of the aspects will not apply to every context.   ***Example:***  A Developer might ask their Product Owner, “*What can you tell me about our client’s desires to add on to this feature in future sprints?*” In this instance, the developer is thinking about the aspect of **Scalability**, and the question may allow some information to bubble to the surface that could aid in modifying the team’s development strategy.  *Don’t forget to decide which user stories will be peer-tested via* ***Shake And Bake****.*  **(continued on back) Card 3B** | *pair-testing-gears.png*Action Items To Limit Chaos ***Pair-Testing & Guided Swarming***     1. Execute ***Shake And Bake*** pair-testing:    1. A 45-90 minute session takes place, post-coding but   pre-commit. Use time-boxing to prevent value deprecation.   * 1. The Tester sits at the Developer’s desk, bringing only a   general outline of questions and potential risks.   * 1. Developer and Tester learn via pair-testing on the Developer’s machine.   2. As bugs are found, the Developer fixes them in-session, refreshes the product, and pair-testing continues.   3. Once the session reaches its natural conclusion, the normal sprint workflow continues. Feature complexity will determine if this can take the place of typical testing sessions.  1. Test Strategies: Testing leaders can use these to guide decentralized testing activities, such as swarming.   **(continued on back) Card 7B** |
| *question-clouds.png*Visualize Testing! ***Create Product Outlines that Lead to Test Strategies***     1. Install ***XMind***, a free mind-mapping tool. 2. Open the ***HTSM Starter Template*** in XMind. 3. Begin the process of typing any and all questions that come to mind when you think about the items that are being developed. 4. After you reach a point where the questions are no longer flowing naturally, use these two explicit models to further inform your thinking:    1. ***Risk Analysis Heuristics (for Digital Products)****.*    2. HTSM’s ***Quality Criteria*** and ***Test Techniques*** nodes.   ***Remember:***  Models like the ones suggested here, ***Risk Analysis Heuristics*** and ***HTSM***, are just a few of the proven tools that already exist in the larger community. Be sure to modify these and seek out others that address your team’s context.  **(continued on back) Card 4B** | 4-icons.pngPeople Trump Process… ***…And a Smart Process Trumps Tradition***     1. **Release Checklist**:    1. Put together a list of high-visibility, high-risk, and chronically buggy product areas within your team’s purview.    2. Convert this list into sanity checks (e.g. A Product Outline) and smoke tests (e.g. A Risk Testing Strategy)    3. Integrate these tests into the Release process (may include automation).    4. These are documented in a way that is sufficient enough for *anyone* from the team to execute them. Deciding between a heavily scripted or lean conceptual format is at team discretion.    5. As usual, context is king, and some releases may not require this. 2. **Reversion Hypotheticals:**    1. Format: *“If ‘x‘ happens, then ‘y’ are the risks to the customer, so we we recommend reverting code commits related to story ‘z’ using the following process…”*   **(continued on back) Card 8B** |
| *collab.png*Intentionally Transparent Strategy ***Iterative & Collaborative Formation***     1. Share the outline or mind-map in the team’s communication   tool (e.g. HipChat team room) to gain insight from others.   1. Take 10 minutes to walk through the outline/strategy with a   Developer and to get their input.   1. Engage the Product Owner in a strategy review session to explore areas of risk that matter to them. 2. Determine if additional bandwidth will be needed to test this feature within the time allotted. 3. Advocate for **Testability**, by consulting with the Developer to see if they could build something to aid in your testing. 4. Update your team’s test strategy with the information you have learned. 5. Repeat the above process as needed to satisfy the priorities laid out by Product Management.   **(continued on back) Card 5B** | *loop-house.png*Close The Loop, Bring It Home ***Lean Prod Testing, Focused Informant, Compelling Story***     1. **Lean Production Testing**: By this point your testing strategy has generated a high confidence level, such that adequate, yet minimal testing is appropriate for the Production environment. Utilize tools such as the ***Release Checklist***. 2. **You Do Not Assure Quality**: This is the job of the company, not the individual. Inform on risk, then step aside so product management can leverage the crowd’s wisdom to make product decisions. Gatekeeping is not within the role of testing. 3. **Testing Story**: Be prepared to tell a compelling story about the testing that was done, by focusing on three main areas: The status of the product, the testing methodologies used, and the quality of the testing. 4. **Team Retro**: Re-poll your team members to close the learning loop and tie off any loose ends. (e.g. *“As a tester, what can I do to make your job easier?”)*   **(continued on back) Card 9B** |

|  |  |
| --- | --- |
| Quarterly Planning Each Release Train has a planning objective that rolls up to business goals. Teams can use models and other heuristics to generate more informed story estimates. Key takeaways:  * Use Guideword Heuristics (rules of thumb for making decisions)   + From HTSM use the ***Product Elements*** and ***Project Environment*** nodes to make more educated judgements. * Use Mnemonics (a character pattern that aids in recall).   **(continued on back) Card 1A** | C:\Users\croberts\Desktop\Sharepoint Docs To Share\RefCards-FRONT.bmp |
| Resources Here are links to some of the models, mnemonics and other tools discussed throughout this reference card pack: Links:  * X-Mind Mind Mapping Tool: **xmind.net/download** * HTSM Testing Model (PDF): **goo.gl/4pfGqA** * HTSM Testing Model (X-Mind): **goo.gl/VTLWjy** * HTSM Starter Template (X-Mind): **goo.gl/YAYwVJ** * Shake and Bake Peer-Testing Model: **goo.gl/48j9NF** * Useful Testing Mnemonics/Models: **goo.gl/FY4QBq** * Risk Analysis for Digital Products (PDF): **goo.gl/BTrgRH** * Risk Analysis for Digital Products (X-Mind): **goo.gl/u4KKpy** * FEW HICCUPPS Consistency Model: **goo.gl/VPtwMY** * Heuristics of Testability: **goo.gl/eDZaUK** | Notes |

|  |  |
| --- | --- |
|  |  |
|  |  |

|  |  |
| --- | --- |
| RefCards-END.png | ../../../../../Desktop/Project_Management-512.pngA Good Start ***Test strategizing begins now!***   1. During Quarterly Planning, review the ***Product Elements***   *and* ***Project Environment*** nodes from HTSM.   1. Use the child nodes from HTSM to fuel more informed discussions and fill gaps in the team’s thinking. Treat this as a guide rather than a checklist, since some of the aspects will not apply to every context.   ***Example:***  A Director might ask a scrum team, “How do we know if these numbers are accurate?” to which a Product Owner might reply, “Part of our process involved the use of HTSM, and explicit model that provides ***Project*** and ***Product*** variables that help inform our thinking. While there are always unknowns, this allowed us to decrease that gap so we can be more confident about our estimations.”  **(continued on back) Card 1B** |
|  |  |



|  |  |
| --- | --- |
|  |  |
|  |  |