### Team Agility Scorecard

<table>
<thead>
<tr>
<th>Objectives</th>
<th>Roles, Mindsets, Events, Artifacts</th>
<th>Product owner role</th>
<th>Team Agility Coach role</th>
<th>Score</th>
<th>Score</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Have a clear reason for the team to go to understand what work needs to be done and why. Responsibilities include:</td>
<td>Roles in the team to answer questions</td>
<td>Product owner role is not defined</td>
<td>No one is doing this role or it’s being done by several people</td>
<td>10</td>
<td>8</td>
<td>15</td>
</tr>
<tr>
<td>Have a coach for the team to assist in improving their definition of workflow and in following it. They are responsible for liaising between the team and the rest of the organization.</td>
<td>Team Agility Coach role is the extent possible</td>
<td>Team Agility Coach role is not doing it or it’s being done by several people</td>
<td>No one is doing this role or it’s being done by several people</td>
<td>10</td>
<td>5</td>
<td>14</td>
</tr>
</tbody>
</table>

### Mindsets

- **1. Have all of the capabilities needed to create value:**
  - 3a Cross-functional teams: Have all of the capabilities needed to create value quickly in a predictable, sustainable and high quality fashion.
  - 3b Create true teams to focus on the team's purpose. They create the team's scope and work in a manner that defines when stories have completed and are not being planned.

### Events / Ongoing

- **4. Keep work within capacity. Lower cycle time:**
  - 4a Development team: Work is started in an ad hoc manner. Generally each person on the team starts work on a separate story when a task is ready. When starting a task people look to see how that story will contribute to bigger picture.
  - 4b Definition of Ready: DoR and DoD are defined, but not commonly used.
  - 4c Definition of Done: Planning is done but it's mostly guesswork.
  - 4d Daily Standups: Daily standups are being done but are not being done.

### Artifacts

- **5. Ensure working on the right thing and don’t start until ready:**
  - 5a Focus on finishing / Managing Work in Process: Team members are just trying to get their job done. The workflow includes some indication of whether stories have had acceptance criteria defined.
  - 5b Definition of Workflow: There is no defined workflow besides “on backlog”, “in process”, “done”.
  - 5c Sprint Planning: Daily standups are being done but from a team perspective. so that “3 done” activities are happening daily.
  - 5d Communication, collaboration, ability to pivot, remove blockers: Daily standups are being done but from a team perspective. so that “3 done” activities are happening daily.

### Quality of Cadence

- **6. Get feedback from stakeholders and product owners and make any necessary corrections on direction:**
  - 6a Product Demonstration: No regular demonstrations are done.
  - 6b Managing Impediments: Impediments are mostly ignored and just treated as is.
  - 6c Use for improving the methods of the team: Retrospections are done but not being done.
  - 6d Ensure backlogs are ready when a sprint or cadence starts: Product backlog is not ready and work is not visible. Work is not visible. There is no organized list of work to be done.

### Visibility of Work (team backlog)

- **7. Create a plan to improve focus and help people see what is happening to enhance dependency management:**
  - 7a Sprint Planning: Stories are mostly completed but cadences are defined but not notified.
  - 7b Communication, collaboration, ability to pivot, remove blockers: Daily standups are being done but from a team perspective. so that “3 done” activities are happening daily.

### Mindsets

- **8. Communication, collaboration, ability to pivot, remove blockers:**
  - 8a Sprint Planning: Daily standups are being done but from a team perspective. so that “3 done” activities are happening daily.
  - 8b Communication, collaboration, ability to pivot, remove blockers: Daily standups are being done but from a team perspective. so that “3 done” activities are happening daily.

### Cadences for readying work prior to being done

- **9. Ensure backlogs are ready when a sprint or cadence starts:**
  - 9a Product Demonstration: No regular demonstrations are done.
  - 9b Managing Impediments: Impediments are mostly ignored and just treated as is.
  - 9c Use for improving the methods of the team: Retrospections are done but not being done.
  - 9d Product backlog: There is no product backlog, but teams are just given work on an ad hoc and urgent basis.

### Visibility of work (team backlog)

- **10. Ensure backlogs are ready when a sprint or cadence starts:**
  - 10a Product Demonstration: No regular demonstrations are done.
  - 10b Managing Impediments: Impediments are mostly ignored and just treated as is.
  - 10c Use for improving the methods of the team: Retrospections are done but not being done.
  - 10d Product backlog: There is no product backlog, but teams are just given work on an ad hoc and urgent basis.

### Cadences for readying work prior to being done

- **11. Get feedback from stakeholders and product owners and make any necessary corrections on direction:**
  - 11a Product Demonstration: No regular demonstrations are done.
  - 11b Managing Impediments: Impediments are mostly ignored and just treated as is.
  - 11c Use for improving the methods of the team: Retrospections are done but not being done.
  - 11d Product backlog: There is no product backlog, but teams are just given work on an ad hoc and urgent basis.

### Visibility of work (team backlog)

- **12. Use visual controls to: 1) show rate of completion of work – eg burn down charts:**
  - 12a Story size, estimation velocity: Stories that are almost ready to be worked on or committed to if using Burndown / Burnup Charts: No reports that show what is being done are available.
  - 12b See if stories are being worked in proper order: 1 of these reports is available.
  - 12c See time taken for each step (eg CFD): 2 of these reports are available.
  - 12d Use type of work being done: 3 of these reports are available.

### Cadences for readying work prior to being done

- **13. Get feedback from stakeholders and product owners and make any necessary corrections on direction:**
  - 13a Product Demonstration: No regular demonstrations are done.
  - 13b Managing Impediments: Impediments are mostly ignored and just treated as is.
  - 13c Use for improving the methods of the team: Retrospections are done but not being done.
  - 13d Product backlog: There is no product backlog, but teams are just given work on an ad hoc and urgent basis.

### Visibility of work (team backlog)

- **14. Use visual controls to: 1) show rate of completion of work – eg burn down charts:**
  - 14a Story size, estimation velocity: Stories that are almost ready to be worked on or committed to if using Burndown / Burnup Charts: No reports that show what is being done are available.
  - 14b See if stories are being worked in proper order: 1 of these reports is available.
  - 14c See time taken for each step (eg CFD): 2 of these reports are available.
  - 14d Use type of work being done: 3 of these reports are available.
Copyright (c) Net Objectives.

Things missing

This scorecard is still being built. The following are comments we've received on things to add. There is a tradeoff between completeness and degree of complicativeness. Comments welcome.

MBI mindset

Objective Based Thinking

measure of team flow behaviors (e.g. none - some pairing - true pair programming - business people working hand in hand with tech developers at task level - mob programming)

measure of individual flow - or a practice that encourages/supports Flow (e.g. https://www.amazon.com/Flow-Psychology-Experience-Perennial-Classics/dp/0061339202)

This IS about SW Dev - so lets talk about storage of artifacts (code Repo)

Automated Test

DoD ready must include test specs

MBI mindset

Objective Based Thinking

measure of team flow behaviors (e.g. none - some pairing - true pair programming - business people working hand in hand with tech developers at task level - mob programming)

measure of individual flow - or a practice that encourages/supports Flow (e.g. https://www.amazon.com/Flow-Psychology-Experience-Perennial-Classics/dp/0061339202)

This IS about SW Dev - so lets talk about storage of artifacts (code Repo)

Continuous integration and at the end of the sprint/cadence demonstrable software gets

Automated Test

None - no concept of test as first class code artifact --- test

21st C definition

(http://agilecomplexificationinverter.blo

1st century- definition-of-test.html) --- everything under our control is automated and one button deployable
<table>
<thead>
<tr>
<th>Cell</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>T2</td>
<td>Self-score each row from 1 (big batch) to 15 (fast, flexible, flow)</td>
</tr>
<tr>
<td>V2</td>
<td>Self-score each row from 1 (big batch) to 15 (fast, flexible, flow)</td>
</tr>
<tr>
<td>C13</td>
<td>See Vish note on grooming frequency</td>
</tr>
</tbody>
</table>