

DELIVERABLE ROADMAP: UMA FERRAMENTA DE PLANEJAMENTO PARA O DESENVOLVIMENTO LEAN DE PRODUTOS

Mayara Silvestre de Oliveira (mayarasilvestredeoliveira@gmail.com) - Departamento de Engenharia Mecânica / Centro Tecnológico / Universidade Federal de Santa Catarina

Laryssa Menegazzo Soares Antunes (laryssa.antunes@polo.ufsc.br) - Universidade Federal de Santa Catarina

Fernando Antônio Forcellini (fernando.forcellini@ufsc.br) - Universidade Federal de Santa Catarina

Jaime Andrés Lozano Cadena (jaime@polo.ufsc.br) - Universidade Federal de Santa Catarina

Jader Riso Barbosa Junior (jrb@polo.ufsc.br) - Universidade Federal de Santa Catarina

RESUMO

O planejamento é uma das principais etapas no desenvolvimento de produtos, principalmente em produtos inovadores. Durante o desenvolvimento, muitas mudanças ocorrem ao longo do processo, o que é agravado pelo pouco conhecimento sobre o produto. No entanto, as abordagens tradicionais de planejamento, baseiam-se na premissa de que é possível planejar as atividades e seguir o plano até o final do projeto. Por outro lado, o Desenvolvimento Lean de Produtos (LPD), em vez de se concentrar em tarefas e cronogramas, concentra-se em estabelecer metas, entregas e marcos de projeto. Este trabalho tem como objetivo apresentar a aplicação de uma ferramenta para planejamento em ambientes de LPD. A ferramenta foi aplicada no desenvolvimento de um protótipo de refrigerador magnético. Como resultado, a ferramenta promoveu a comunicação, integração e motivação dos desenvolvedores e a criação de uma cultura de foco no valor. Além disso, a ferramenta contribuiu para controlar o afunilamento do espaço de projeto durante a Engenharia Simultânea Baseada em Conjuntos (SBCE).

Palavras chave: *desenvolvimento lean de produtos; engenharia simultânea baseada em conjuntos; planejamento baseado em responsabilidade*

1. INTRODUÇÃO

O desenvolvimento de produtos propicia um ambiente com incertezas e mudanças em relação ao que foi planejado. No entanto, nas abordagens tradicionais, chamadas *Point-Based Design* (PBD), os planos são orientados para tarefas e *gates* de projeto. Esses métodos de planejamento consideram que o futuro é previsível e que se sabe exatamente o que fazer para atingir as metas de desenvolvimento (ROTHER; AULINGER, 2017). Também conhecido como Planejamento Baseado em Tarefas. Por outro lado, o Desenvolvimento Lean de Produtos (LPD) e sua Engenharia Simultânea Baseada em Conjuntos (SBCE) adotam outra perspectiva de planejamento. O foco não se dá em tarefas e cronogramas, mas sim em entregas, metas e marcos de projeto. Entre os eventos de integração associados à SBCE, os subsistemas e suas equipes pesquisam e definem seus espaços de projeto (ou conjuntos de soluções) por meio de ciclos de aprendizado, com o objetivo de afunilar as alternativas de solução para o produto (SCHIPPER; SWETS, 2010, WARD, 2011). As equipes de cada subsistema planejam e gerenciam seus próprios ciclos de aprendizado com base no método científico formalizado pela abordagem Planejar-Fazer-Controlar-Agir (PDCA). Dessa forma, entregas e metas são estabelecidas para o desenvolvimento, ao invés de ações.

O uso de ferramentas de gestão visual tem sido o primeiro passo para uma implementação do LPD em muitas empresas. As entregas do projeto são definidas e ilustradas em um painel de planejamento, de forma que possam ser avaliadas frequentemente. Entretanto, existem poucas publicações sobre a gestão visual e planejamento no LPD (LINDLOF; SODERBERG, 2011). Portanto, muitas questões permanecem sem solução. Quais são os métodos, abordagens e ferramentas que podem ajudar a equipe de desenvolvimento no planejamento em ambientes LPD? Como o plano funciona junto com o processo de afunilamento de alternativas de solução da SBCE?

Com base nesse cenário, este trabalho tem como objetivo demonstrar a aplicação da ferramenta de planejamento *Deliverable Roadmap* (DR) em ambientes LPD, conforme inicialmente proposto por Mascitelli (2011). A ferramenta foi aplicada no desenvolvimento de um refrigerador magnético. Como resultado, a DR forneceu um instrumento para prever os impactos dos problemas e criar uma cultura de foco no valor. Além disso, auxiliou no planejamento dos eventos de integração e no controle do afunilamento do espaço de projeto da SBCE. Concluiu-se que a DR contribui para o planejamento em LPD, auxilia no controle da SBCE de forma simples e

fornece um panorama visual de todo o projeto o que permite que a equipe de desenvolvimento mantenha o foco no valor.

2. DESENVOLVIMENTO DE PRODUTOS LEAN

O LPD é a aplicação do *lean* em ambientes de desenvolvimento (LEON; FARRIS, 2011). No LPD é fundamental definir o que é valor para o consumidor. Esse valor deve ser entendido, compartilhado e desdobrado em objetivos para todos os níveis e equipes que fazem parte do processo de desenvolvimento (LIKER; MORGAN, 2006). Toda atividade que não agrega valor ao cliente é considerada desperdício, que pode ser classificado como: superprodução, espera, transporte, processamento, estoque, movimentação e retrabalho. Entre as causas para os desperdícios estão a falta de liderança, falta de planejamento e gestão, pouca interação entre as equipes de desenvolvimento, não utilização dos conhecimentos prévios e burocracia (OPPENHEIM, 2004).

O LPD tem seus fundamentos em princípios como: foco em valor, SBCE, fluxo, Engenheiro Chefe (EC) e equipes multifuncionais de especialistas, melhoria contínua, padronização, adaptação de tecnologias a pessoas e processos e gestão visual (WARD, 2011, LIKER; MORGAN, 2006, HOPPMANN *et al.*, 2011). O princípio mais importante é o foco no valor. Para garantir a entrega de valor, o LPD possui como estratégia trabalhar várias alternativas de solução ao mesmo tempo, estudando *trade-offs* de projeto, usando prototipagem rápida e gerenciamento visual, para afunilar gradativamente essas alternativas pelo princípio da SBCE (RAUDBERGET, 2010).

Para que seja possível realizar esse afunilamento gradativo, o produto deve ser dividido em subsistemas, para os quais são alocados times multifuncionais responsáveis pelo seu desenvolvimento. Através de um fluxo cadenciado de desenvolvimento, desdobramento do valor e coordenação de um EC, que exerce o papel de líder integrador de todas as equipes dos subsistemas, a SBCE é realizada (WARD, 2011).

2.1 Engenharia Simultânea Baseada em Conjuntos

A SBCE é a estratégia de convergência de soluções adotada no LPD. Durante o desenvolvimento, existem muitas soluções possíveis para resolver o problema do cliente. Em vez de escolher uma solução, aquela considerada a melhor, o LPD possui a premissa de que a equipe de

desenvolvimento não tem conhecimento suficiente para tomar essa decisão. Mesmo que pesquisas, experimentos e testes com essas possíveis soluções sejam realizados, ainda se considera arriscado escolher uma como sendo a melhor solução. Portanto, a SBCE trata este problema desenvolvendo todas as soluções paralelamente e à medida que elas se mostram inadequadas (não atendem aos requisitos do cliente, não são compatíveis com outros subsistemas ou são inviáveis) elas são eliminadas. Isso é chamado de processo de afunilamento do espaço de projeto (WARD, 2011).

Esse processo é gerenciado por eventos de integração com todos os subsistemas, EC e partes interessadas, com o objetivo de integrar desenvolvedores, construindo consenso e afunilando o espaço de projeto. Além disso, durante esses eventos, importantes questões de projeto são respondidas, com base nas curvas de *trade-off*, requisitos de clientes, experimentos e resultados de testes (SCHIPPER; SWETS, 2010, WARD, 2011).

Entre os eventos de integração, ocorrem períodos de pesquisa e descoberta nos subsistemas. Quando necessário, a equipe de cada subsistema planeja e executa experimentos para enfrentar os obstáculos e assim, obter o conhecimento necessário para desenvolver as soluções baseando-se nos ciclos do PDCA, a fim de alcançar as metas dos eventos de integração e os objetivos do desenvolvimento (WARD, 2011). Dessa forma, o LPD se fundamenta na abordagem de planejamento baseado em responsabilidade, na qual o responsável pelo planejamento de atividades é o próprio desenvolvedor, que é cobrado apenas por resultados. Planeja-se resultados e não atividades. Entre os eventos integradores, as equipes planejam as ações a serem realizadas em ciclos curtos de PDCA para que possam atingir os resultados.

2.1 Gestão Visual e Planejamento no Desenvolvimento Lean de Produtos

A gestão em ambientes LPD é baseada no uso de práticas de gerenciamento visual, que permitem o acompanhamento e controle de seus processos, a captura do estado atual das atividades dos subsistemas e o planejamento em tempo real das ações futuras (EPPLER; BURKHARD, 2007). No desenvolvimento de produtos inovadores é comum a quantidade excessiva de informações que devem ser gerenciadas simultaneamente, o que pode levar à sobrecarga. Para evitar esse problema, o gerenciamento visual pode ser utilizado (PARRY; TURNER, 2006, TAXÉN; LILLIESKÖLD, 2008). O uso de gráficos, cores e símbolos fornece informações claras e exatas para os subsistemas, líderes e EC. Recomenda-se que as informações sejam fornecidas a partir de apenas uma fonte e atualizadas por apenas uma pessoa, para evitar versões conflitantes e não

atualizadas (WARD, 2011). Estruturas de gerenciamento visual no LPD sintetizam o status relacionado a um dado projeto, facilitando assim a visualização de um cenário de desenvolvimento (MASCITELLI, 2011).

O papel do controle visual no desenvolvimento de produtos é um assunto bastante abordado na literatura (LINDLOF; SODERBERG, 2011, MASCITELLI, 2011, TAXÉN; LILLIESKÖLD, 2008, OLAUSSON; BERGGREN, 2010, LINDLÖF, 2014, KIM *et al.*, 2018). Entre os benefícios do gerenciamento visual, destacam-se a sua contribuição para a integração do conhecimento, trabalho em equipe, simplificação da gestão de projetos complexos e envolvimento de desenvolvedores na tomada de decisões (LINDLOF; SODERBERG, 2011, LIKER; MORGAN, 2006, OLAUSSON; BERGGREN, 2010, LINDLÖF, 2014, KIM *et al.*, 2018). Além disso, o uso de ferramentas de gestão visual no planejamento torna os problemas do projeto evidentes, permitindo a adoção rápida de contramedidas (LIKER; MORGAN, 2006).

Painéis são as principais estruturas usadas para comunicar informações. Por meio deles, informações são trocadas entre os desenvolvedores, principalmente durante as reuniões do projeto. O planejamento pode ser feito alocando notas adesivas no painel com o uso de cores para realçar certas informações e para fornecer evidências de problemas e riscos. Com a atualização contínua do status do projeto no painel, é possível visualizar a carga de trabalho e equilibrá-la entre a equipe, evitando problemas futuros e reduzindo os riscos associados ao desenvolvimento (LINDLOF; SODERBERG, 2011).

O gerenciamento visual também pode ser feito por meio de métodos de planejamento baseados em software. Entretanto, alguns autores argumentam que esses métodos nem sempre resolvem a sobrecarga de informações do desenvolvimento de produtos (LINDLOF; SODERBERG, 2011, TAXÉN; LILLIESKÖLD, 2008, PARRY; TURNER, 2006). Quando existe muita complexidade associada ao desenvolvimento, um grande esforço é demandado para gerenciar o projeto. Nesse caso, o uso de software pode se tornar muito complexo e demorado, impondo altas demandas às capacidades de processamento de informações de uma organização. Isso implica em uma redução da comunicação entre equipes (LINDLOF; SODERBERG, 2011, TAXÉN; LILLIESKÖLD, 2008, PARRY; TURNER, 2006). Alguns autores afirmam que é muito benéfico que a gestão visual imponha limitações físicas para a informação, como em painéis, por exemplo, pois isso leva a um na qualidade e relevância dos dados apresentados (LINDLOF; SODERBERG, 2011, PARRY; TURNER, 2006).

3. A DELIVERABLE ROADMAP PARA O DESENVOLVIMENTO LEAN DE PRODUTOS

A *Deliverable Roadmap* (DR) é uma ferramenta de gestão visual, na qual todas as entregas necessárias para fornecer o valor ao cliente são detalhadas. O objetivo da ferramenta é permitir o planejamento completo de todas as entregas do projeto desde as suas fases iniciais até o lançamento. Além disso, ela permite que essas entregas sejam distribuídas em vários níveis, sendo subsistemas, equipes ou até desenvolvedores. A DR possui alguns elementos chave, conforme é apresentado na Figura 1, como o uso de marcos de projeto, linha do tempo de desenvolvimento, raias de tempo, raias de responsabilidade e as entregas.

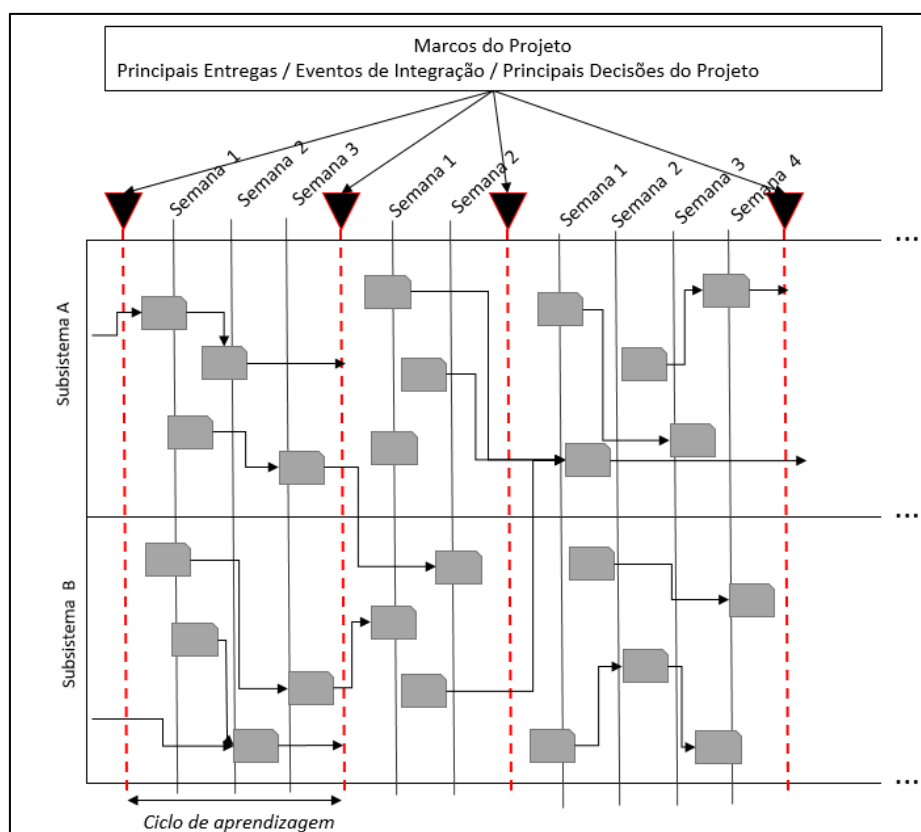


FIGURA 1 – Deliverable Roadmap

Os marcos de projeto (triângulos pretos destacados na Figura 1), correspondem a todas as datas em que algum acontecimento importante deve ocorrer. Dentre eles, destacam-se entregas chave, eventos de integração e decisões cruciais para o projeto. As raias de tempo são as linhas verticais, que dividem a linha do tempo em intervalos regulares, que podem corresponder a semanas,

meses, semestres ou até anos. A linha do tempo do desenvolvimento corresponde a todas as raias de tempo da DR, onde todas as entregas do projeto devem ser distribuídas.

O uso de raias de tempo permite definir ciclos de aprendizagem para os subsistemas, equipes ou desenvolvedores. Dessa forma, é possível que eles planejem suas atividades no período correspondente às raias para que possam cumprir suas entregas. As raias de responsabilidade correspondem as linhas horizontais da DR. Dentro dessas raias estarão todas as entregas de um subsistema, equipe ou desenvolvedor.

Recomenda-se que as entregas sejam alocadas na forma de cartões, conforme é detalhado na Figura 2. É importante que as informações registradas nos cartões sejam completas, precisas e não ambíguas. No caso de as raias de responsabilidades estarem referidas à subsistemas ou equipes, é recomendado designar um responsável em particular para cada entrega, mesmo que várias pessoas estejam envolvidas.

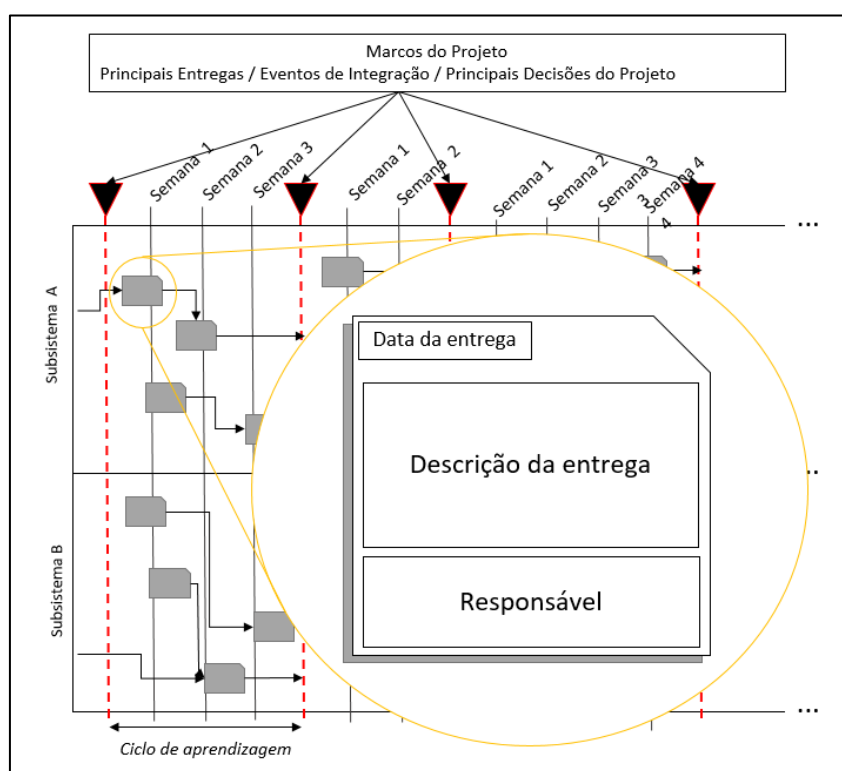


FIGURA 2 – Detalhe das entregas na Deliverable Roadmap

A construção da DR deve ser feita em conjunto com todos os envolvidos, principalmente os responsáveis apontados nas raias de responsabilidade. Além disso, é importante que o responsável pelas entregas tenha clareza sobre o que é esperado para a entrega, quando ela deve ocorrer e por que ela deve ser cumprida. Deve-se buscar também, relacionar as entregas com o valor esperado ao final do desenvolvimento, de forma a evitar entregas desnecessárias que não contribuam para a entrega de valor. Recomenda-se que a DR seja periodicamente revisada e atualizada, à medida que se adquire mais conhecimento acerca do desenvolvimento e do produto e com isso se tenha novas perspectivas quanto às entregas necessárias. A relação de precedência também deve ser detalhada, a fim de fornecer uma visão da cadeia de entregas e do caminho crítico do projeto.

A DR não deve fornecer um plano de ação para os desenvolvedores e sim metas e entregas. Os responsáveis devem planejar suas próprias atividades, com base na abordagem PDCA, a fim de realizar as suas respectivas entregas na DR. Recomenda-se colocar primeiramente as principais entregas e então, desdobrá-las em entregas menores. Além disso, o planejamento deve ser iniciado do final do projeto em direção ao presente. Dessa forma é possível obter uma visão do valor entregue e dos prazos do projeto.

A clareza sobre o que deve ser entregue diminui quando se trata de horizontes de tempo distantes. Por isso, é possível alterar o detalhamento do plano de acordo com as raias de tempo. Raias semestrais permitem um nível menor de detalhamento do que raias semanais. Por isso, em horizontes de tempo próximos do atual recomenda-se o uso de raias semanais. A medida que o planejamento é feito para o longo prazo, recomenda-se que se use intervalos maiores.

A DR pode ser utilizada para distribuir regularmente ao longo do tempo os eventos de integração do projeto e atrelar a eles entregas cruciais para o afinilamento de alternativas de solução como a entrega de curvas de *trade-off*, resultados de testes de falha ou experimentos com materiais para determinados subsistemas. Essas entregas e eventos integradores podem ser também distribuídas para atender a importantes marcos de projeto como a construção de um protótipo que integre os subsistemas. Dessa forma, é possível gerenciar o processo de afinilamento da SBCE.

Entre os eventos de integração irão ocorrer os chamados ciclos de aprendizagem, conforme é apresentado na Figura 1. Nesses ciclos, para problemas em que não se conhece a solução os membros dos subsistemas ‘aprendem’ sobre o produto por meio de experimentos e testes. É

possível planejar as entregas que consolidam esse aprendizado como as curvas de *trade-off*, por exemplo.

4. RESULTADOS E DISCUSSÃO

A DR foi aplicada no desenvolvimento de um refrigerador magnético, cujo prazo final de desenvolvimento corresponde a quatro anos. Entretanto, nos primeiros dois anos, o objetivo foi a comprovação da viabilidade técnica por meio do desenvolvimento de um protótipo. O EC, inicialmente tentou utilizar técnicas tradicionais de planejamento, como cronogramas em gráficos de Gantt e estruturas analíticas de projeto. Entretanto, essas técnicas acarretavam em dificuldades para o gerenciamento das várias equipes que compunham o time de desenvolvimento.

Foi relatado pelo EC que a principal dificuldade encontrada era acompanhar o trabalho de cada membro da equipe de maneira sistemática. Além disso, existia a necessidade de visualizar todo o desenvolvimento, o que com as técnicas adotadas não era possível. Com base nesse cenário, os autores propuseram o uso da DR para planejar o projeto. A DR foi utilizada para planejar e gerenciar o desenvolvimento do protótipo. A primeira DR do refrigerador magnético é apresentada na Figura 3 de forma miniaturizada por questões de confidencialidade associada ao desenvolvimento do produto.

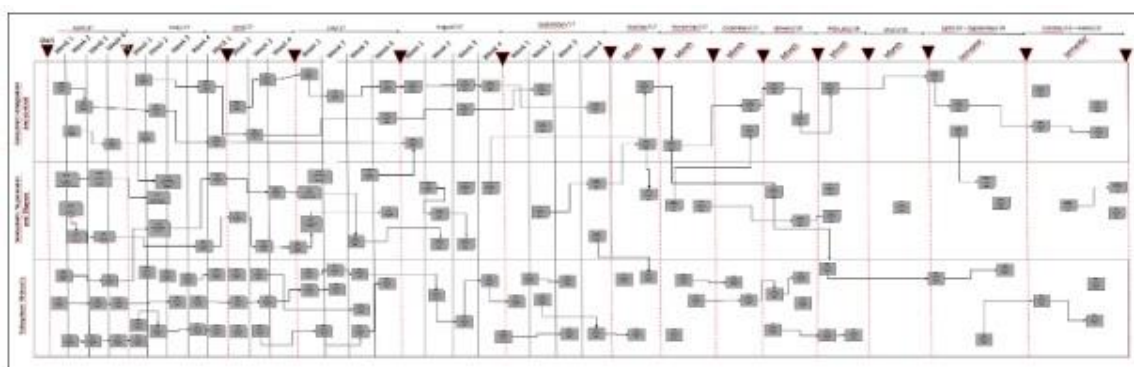


FIGURA 3 – Primeira Deliverable Roadmap do refrigerador

O desenvolvimento da DR foi liderado pelo EC, que estabeleceu os principais marcos de projeto e entregas críticas. Com base nisso, as equipes dos subsistemas realizaram o desdobramento das entregas críticas em entregas menores dentro das suas raias de responsabilidade. Como o produto era composto de três subsistemas, por esta razão, utilizou-se três raias de responsabilidade na DR do refrigerador. Cada desenvolvedor assumiu a responsabilidade de suas entregas

que estavam registradas na DR em cartões, cujo formato é apresentado na Figura 2. Devido às incertezas associadas ao desenvolvimento, principalmente em períodos mais próximo ao final do período de dois anos, decidiu-se que as raias de tempo nos primeiros quatro meses seriam semanais, nos oito meses seguintes mensais e no último ano, semestrais. Para o último ano foram definidas apenas as principais entregas.

Primeiramente, o EC atribuiu os principais marcos de projeto, como os prazos finais para decisões importantes e entregas para o patrocinador do projeto. A partir disso, as principais entregas foram definidas. Em seguida, em uma reunião com as equipes responsáveis por cada subsistema, as raias de responsabilidade foram preenchidas, do final do projeto até o início, por meio do desdobramento de entregas maiores. Além disso, em um evento formal a equipe apresentou a função e objetivo de cada subsistema, bem como as principais entregas.

A DR foi elaborada, inicialmente em um papel pardo e as entregas foram alocadas usando notas adesivas. Após, uma versão online foi compartilhada com todas as equipes, utilizando o software Microsoft Visio®. Entretanto, à medida que a DR era atualizada, versões conflitantes do arquivo ocorreram. Por isso, migrou-se para o draw.io®, pois ele permite atualizações simultâneas vindas de vários usuários. Periodicamente, a DR foi revisada e não apenas o detalhamento das entregas era refinado, mas também as raias de tempo que antes correspondiam a meses, passaram a ser semanas.

Os eventos de integração e as entregas a serem apresentadas por cada subsistema em cada evento foram definidas na DR. As principais entregas definidas como insumo para os eventos foram curvas de *trade-off*, resultados de simulações, validação de modelos com resultados experimentais, testes e estudo de normas e estado da arte. O objetivo dessas entregas foi adquirir conhecimento sobre áreas específicas em pontos estratégicos do desenvolvimento e também, possibilitar que se atinjam os objetivos em cada marco de projeto, ou evento integrador.

A DR facilitou o acompanhamento do projeto pelo EC. Houveram alguns atrasos nas entregas, como atrasos no fornecimento de peças e componentes devido a questões de importação e prazo de entrega de fornecedores. Entretanto, foi relatado pelo EC, que a DR possibilitou prever os impactos desses atrasos. Como a DR forneceu um panorama das entregas do projeto, foi possível ver como o atraso de uma determinada entrega afetaria as demais e um rápido planejamento de contramedidas para evitar atrasos importantes pode ser feito. Foi possível também integrar os desenvolvedores na tomada de decisões. As relações de precedência e as raias permitiram que um senso de urgência surgisse sempre que alguma entrega não era cumprida no prazo.

É importante ressaltar que a construção e atualização da ferramenta em meios físicos (papel e notas adesivas), de forma dinâmica, permitindo que os desenvolvedores realocassem entregas facilmente. Quando a atualização da DR ocorreu utilizando um software, a visualização da linha do tempo completa ficou comprometida devido ao longo horizonte de tempo da DR e a atualização das entregas foi dificultada.

Quando a DR foi disponibilizada no software Visio[®], os conflitos de versão ocorreram conforme preconizado por Lindlof e Soderberd (2011). Quando revisões da DR eram feitas por mais de um usuário ao mesmo tempo, essas versões apareciam dificultando manter um arquivo único e atualizado. Por isso, foi proposto o uso de um meio eletrônico que permitisse alterações oriundas de vários usuários simultaneamente. O aplicativo draw.io[®] se mostrou adequado pois permitiu que todos os desenvolvedores pudessem acompanhar e atualizar suas linhas do tempo simultaneamente.

Nos métodos tradicionais de planejamento, nem sempre é possível entender a importância de uma tarefa para o desenvolvimento como um todo ou para outro subsistema. Foi percebido que, a DR por viabilizar o planejamento por responsabilidades e engajar os desenvolvedores na elaboração do planejamento, motivou a equipe e aumentou sua integração. Dessa forma, a DR se mostrou, também, uma ferramenta de integração. Foi percebido que, com as informações da DR, foi possível que as equipes de diferentes subsistemas se unissem durante os experimentos para aprender sobre os parâmetros do produto de forma integrada.

Observou-se também que o planejamento baseado em responsabilidades contribuiu para evitar desperdícios no desenvolvimento. Foi possível visualizar a cadeia de entregas na DR, o andamento do projeto e desvios do plano. Além disso, as entregas foram planejadas para entregar valor, o que permite o desenvolvimento de uma cultura de foco no valor e não na tarefa. Os desenvolvedores criaram uma rotina de organização e planejamento de seu trabalho diário que levava em conta não apenas as entregas, mas também a qualidade esperada.

5. CONCLUSÃO

O uso da DR foi demonstrado no desenvolvimento de um protótipo de refrigerador magnético. Foi observado que a ferramenta forneceu uma visão geral do projeto para o EC e desenvolvedores, e eles puderam gerenciar o desenvolvimento usando a ferramenta proposta. Além disso, foi possível prever problemas e propor contramedidas, promover a integração e gerir as incertezas. Para tanto, concluiu-se que a ferramenta proposta contribui para o planejamento no LPD.

Conclui-se também, que a DR permite controlar o afunilamento do espaço de projeto e os experimentos associados à SBCE. A DR foi usada para planejar os eventos de integração, suas saídas e entradas. Além disso, testes, experimentos e desenhos de curvas de *trade-off* foram planejados usando a ferramenta proposta.

O foco no valor, e não nas tarefas, foi um dos resultados observados da DR. Como os desenvolvedores puderam compreender a importância e os impactos de cada entrega, houve um aumento do engajamento e motivação. Percebeu-se que as equipes estavam focadas no valor e objetivos do projeto. Com base nisso, concluiu-se que a DR contribui para a motivação durante o desenvolvimento. Além disso, ajuda a manter o foco no valor e evita desperdícios durante o projeto.

6. REFERÊNCIAS

- EPPLER, M.J.; BURKHARD, R.A. Visual representations in knowledge management: framework and cases. **Journal of knowledge management**, v.11, n.4, p. 112-122, 2007.
- HOPPMANN, J. REBENTISCH, E.; DOMBROWSKI, U.; ZAHN, T. A framework for organizing lean product development. **Engineering Management Journal**, v.23, n.1, p. 3-15, 2011.
- KIM, E.; BECKMAN, S. L.; AGOGINO, A. M. Applying Design Roadmapping in New Product Development Education: Insights from Student Design Teams. ASME 2018 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. **American Society of Mechanical Engineers**, 2018.
- LEÓN, H.C.M.; FARRIS, J. A. Lean product development research: Current state and future directions. **Engineering Management Journal**, v.23, n.1, p. 29-51, 2011.
- LINDLOF, L.; SODERBERG, B. Pros and cons of lean visual planning: experiences from four product development organizations. **International Journal of Technology Intelligence and Planning**, v.7, n.3, p. 269-279, 2011.
- LINDLÖF, L. **Visual management: On communication in product development organizations**. Chalmers University of Technology, 2014.
- LIKER, J.K.; MORGAN, J.M. The Toyota way in services: the case of lean product development. **Academy of Management Perspectives**, v.20, n.2, p. 5-20, 2006.
- MASCITELLI, R. **Mastering lean product development: A practical, event-driven process for maximizing speed, profits and quality**. Northridge, CA: Technology Perspectives, 2011.
- OLAUSSEN, D.; BERGGREN, C. Managing uncertain, complex product development in high-tech firms: in search of controlled flexibility. **R&D Management**, v.40, n.4, p. 383-399, 2010.
- OPPENHEIM, B.W. Lean product development flow. **Systems Engineering**, v.7, n.4, 2004.
- PARRY, G.C.; TURNER, C.E. Application of lean visual process management tools. **Production Planning & Control**, v.17, n.1, p. 77-86, 2006.
- RAUDBERGET, D. Practical applications of set-based concurrent engineering in industry. **Journal of Mechanical Engineering**, v.56, n.11, p. 685-695, 2010.
- ROTHER, M.; AULINGER, G. **Toyota Kata Culture: building organizational capability and mindset through kata coaching**. McGraw Hill Professional, 2017.
- SCHIPPER, T.; SWETS, M. **Innovative lean development: how to create, implement and maintain a learning culture using fast learning cycles**. CRC Press, 2012.

- SMITH, P.G. **Flexible product development: building agility for changing markets**. John Wiley & Sons, 2007.
- TAXÉN, L.; LILLIESKÖLD, J. Images as action instruments in complex projects. **International Journal of Project Management**, v.26, n.5, p. 527-536, 2008.
- WARD, A.C. **Sistema Lean de desenvolvimento de produtos e processos**. São Paulo: Leopardo Editora, 2011.