International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2019): 7.583

Exploring the Imperative for Technical Product Managers in Platform Teams: A Position Paper

Srividhya Chandrasekaran

Senior Product Manager, Bedford, MA
Email: *srividhya.chandrasekar[at]gmail.com*

Abstract: ***Product managers play an important role in collaborating with product teams and end users to define requirements, guide product development, and ensure user satisfaction by being the voice of the customer. However, the role of a Platform Product Manager (Platform PM) introduces a unique dimension within product management. Tasked with constructing the foundational systems that empower product teams in feature development, Platform PMs create platforms for both end users and internal teams. Platform product managers are vital to the health of a technology company [1]. This position paper delves into the critical significance of Platform PMs possessing a profound understanding of the technical intricacies inherent in their platform products. As organizations shift with evolving objectives driven by emerging tools and technologies within platforms, the necessity for Product Managers to grasp the fundamentals of the underlying technology becomes increasingly evident. This comprehension is instrumental in addressing challenges related to performance, reliability, uptime, and scalability. The exploration within this paper aims to elucidate the role performed by Platform PMs, assessing the nature of challenges faced by them. A comparative analysis is presented, differentiating between the technical and non - technical aspects of a traditional PM versus a Platform PM. This investigation seeks to define a distinct position on the role of a Platform PM, highlighting its divergence from that of a traditional product manager.***

Keywords: Platform Product Manager, DevOps, Shift Left, End - user, Continuous Integration

1. Introduction

Platform Product management is an emerging field. Platforms can be defined [2] as “the foundational software that application developers use to create end - user solutions”. While regular product managers focus on the requirements for product features, platform product managers focus on the requirements of the various tools, systems, and reusable components necessary for supporting the applications and products built and sold by companies. An example that can be used here is Samsung’s Android phone [3]. While one of the primary products that product managers at Samsung might focus on is the phone itself, all the software that is used by the application’s developers that support building the phone including Operating Systems (OS), Services, etc. fall under platforms. Any tool/service that internal teams within a company build to support the DevOps and ‘Shift left’ practices for example a Pipeline of Continuous Integration/Continuous Deployment or a pushbutton tool using Ansible to build environments faster, also fall under platforms. However, these platforms must be built in methods similar to a product. What this means is that the requirements for these platforms need to be vetted by the internal and external teams that will use the platform as an end user. This makes it vital for the platforms to be built on solid product management and engineering principles and overseen by product management leaders that understand the needs of the end users.

2. A Glimpse into the Typical Product Management Cycle

The below stages form a standard sequence for a product encompassing key phases that guide its development and lifecycle.

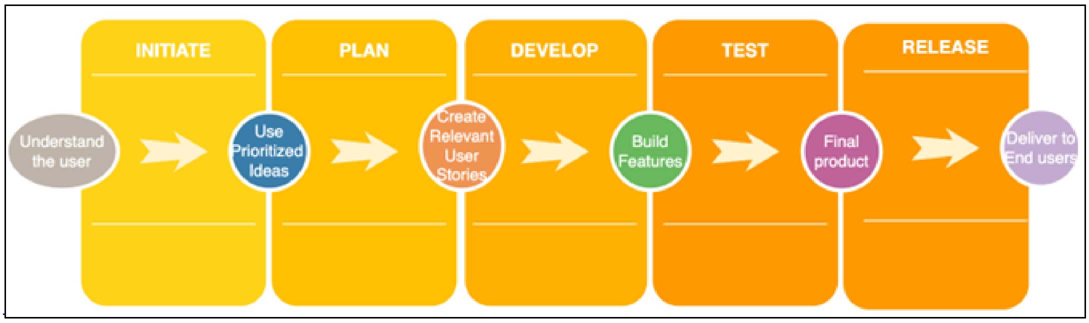

Figure 1: Key phases in Product development

This sequence begins with an initiation or ideation phase where ideas are brainstormed and refined, followed by a planning phase where the ideas are prioritized, estimated, and user stories defined. A necessary part of the Initiate phase is for the Product Manager to work with the end users to understand their needs and keep their user personas in mind in the rest of the phases. The next phase is the development phase where these user stories are developed along with the engineering team by grooming them and unit tested after development. The testing phase involves testing the final feature end to end along with an integrated testing phase where the engineers make sure that the new feature does not

DOI: https://dx.doi.org/10.21275/SR24304114639

break any existing features or systems that it would interact with. The release phase involves deploying and releasing the feature or the product so that the end user can beta test it. The release phase may include a system integration testing before release.

While this is usually the standard process, some engineering teams have started to continuously integrate and deploy features as they build and test it. This helps identify issues ahead of time without waiting for the test or release phase when there is a possibility of critical and major defects being unearthed causing delays in release.

It is crucial to understand that platform products also go through similar phases. But due to the unique nature of Platform Products, there exist a ton of differences. While traditional software products might be standalone applications developed to solve a specific problem for the end user, platform products are entire ecosystems that provide an infrastructure on which multiple services or tools can be built.

The differences also lie in the persona of the end users and the inherent technical nature of the requirements that Platform PMs need to understand to move forward. Unlike a traditional Product Manager who may concentrate on a singular product, a Platform PM's role requires a holistic approach, considering how their platform supports and enhances multiple products and a matrix of users.

3. Differences between the responsibilities of a traditional product manager and a platform product manager

Examining specific instances, we will illustrate the distinctions between a Platform Product Manager overseeing an internal pushbutton tool automating the construction of customer environments and a traditional Product Manager responsible for the oversight and management of an Electronic Health Record (EHR) System.

Table 1: Differentiating product management responsibilities

Traditional Product manager (EHR system)	Platform Product Manager (Push Button tool)
End users are users of the application such as healthcare providers	End users are internal developers, operations engineers, infrastructure engineers
Manages the product backlog by interfacing with the application team and the customers	Manages the product backlog by interfacing with a matrix of teams including dev - ops engineers, infrastructure engineers and customers using the environment that was built
Product Backlog includes tickets for building individual features for the EHR system so end users are not blocked at any time.	Product backlog focuses on tickets for building the Infrastructure as Code tool along with making sure the platform is reliable, scalable and performant so that other teams are not blocked at any time.

The above table indicates that while traditional PMs can be highly functional and a clear understanding of the domain is important, Platform PMs need to be hyper focused on the technical details of the tool as well. While traditional Product Managers may concentrate on the immediate market and product improvements, Platform PMs need to balance short - term goals with the long - term vision for the entire platform.

4. Step - by - step Guide for Platform Product Managers to Launch a Product Platform

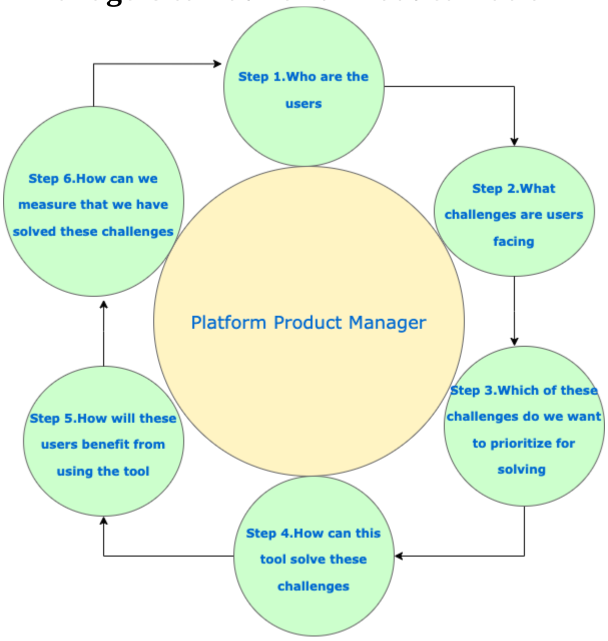
Step 1.Who are the users
Step 2.What challenges are users facing
Step 3.Which of these challenges do we want to prioritize for solving
Step 4.How can this tool solve these challenges
Step 5.How will these users benefit from using the tool
Step 6.How can we measure that we have solved these challenges
Platform Product Manager

Figure 2: Key phases in Product development

a) *Initiate*

- While evaluating a project to build a platform tool, Platform PMs need to understand **who** are the engineers/users using the tool and what are their user persona.
- Understanding the challenges behind the need for the tool is also extremely important so that the tool is built to solve the challenges faced by teams. One example of a consideration a Platform PM could pursue is that if manually building a customer environment is taking 5 - 6 days for an operations engineer, can this tool reduce this time by at least 75%.
- In the example above, while there might be multiple challenges faced by the engineers building the customer environment, the Platform PM needs to ruthlessly prioritize the most important challenges. Prioritizing will reduce scope creep and help the team building the tool have improved focus and lesser distractions.

b) *Plan*

- The next step in the process would be to brainstorm with the team of engineers on how this tool can contribute to solving these challenges and generate ideas

c) *Develop*

In order to develop the right product/tool the team needs to write user stories by understanding deeply how the engineers will use and benefit the tool. While an infrastructure engineer might just need to use the tool enough to ensure that the tool functions smoothly without any permission or space issues, an operations engineer might need to use the tool from start to end in their daily work to build customer environments. As we ‘shift left’ this responsibility could lie with the DevOps Engineers too. While Shifting left is an emerging trend, Platform PMs need to deeply understand dev - ops concepts so they can continuously improve. Another perspective is to understand if this tool needs to have a UI component or if it can be a command line tool only.

d) Test/Release

Once the tool is built and released to the end users, the platform PM is responsible for making sure that the tool is used right and for measuring the outcomes associated with the tool. Some questions they can ask and track usage and metrics related to the tool include

- What % of time has been saved with this automated tool
- Is this tool easy enough to adopt for new users.
- How many users use this tool on a weekly basis
- Are users satisfied with the tool, if not, what can we do better
- Is the tool scalable, reliable and performant

At every stage in the steps above, the Platform PM needs to understand and document the needs of the platform product back into the backlog, communicate the product vision to the scrum team, understand release goals, make decisions with the customer and business in mind and maintain the backlog as well. [4]

5. Conclusion

The findings of the 2019 State of Product leadership report [5] highlight that prioritization skills are the second most important skill within product teams. This need, as revealed in the report, emphasizes the significance of prioritization for effective product management. For Platform PMs, the ability to strategize and prioritize objectively and decisively, coupled with the skill to diplomatically decline requests that may deviate the team from the product's strategic focus, stands out as invaluable. Another gap that was identified by this report indicates lack of alignment between PMs and customer success teams. The report recommends evaluation if customers are retained and customer issues are surfaced on time. This is also important for platform Product managers because if users stop using their tools, teams regress to manual methods and valuable time is lost.

By honing these skills, Platform PMs play a pivotal role in steering their teams towards success. This ensures optimal utilization of the team's resources, allowing them to concentrate on the actual development of the product or tool. The focus and energy of the team is then on creating user stories, adhering to timelines, and delivering top - notch quality and performance.

The absence of a proficient Platform PM can lead to a lack of direction within the team. Without a clear vision and strategic guidance, the team may find itself adrift, resulting in the development of tools that fail to gain user traction. In conclusion, the role of a Platform PM proves indispensable in aligning the team's efforts, fostering a clear vision, and driving the development of products that genuinely meet user needs and expectations.

References

[1] B. de Haaff, “The Curse of the Platform PM, ” *Aha!*, Feb.05, 2015. https: //www.aha. io/blog/the - curse - of - the - platform - product - manager

[2] M. Cagan, “Platform Product Management, ” *Silicon Valley Product Group*, Mar.15, 2005. https: //www.svpg. com/platform - product - management/

[3] *Tiwana, B. Konsynski, and A. A. Bush, “Research Commentary: Platform Evolution: Coevolution of Platform Architecture, Governance, and Environmental Dynamics, ” Information Systems Research, vol.21, no.4, pp.675–687, 2010, doi: 10.2307/23015636.*

[4] “TheScrumTeam, ”*ScrumAlliance*. https: //resources. scrumalliance. org/Article/scrum - team?utm_source=badgecert. com&utm_medium=social&utm_campaign=bdg - sharedigitalbadge&utm_content=scrum - team

[5] The State of Product Leadership 2019, ” *Pendo. io*, Jul.29, 2019. https: //www.pendo. io/resources/the - state - of - product - leadership - 2019/#roles - and - responsibilities