

The App Trail

*How ideas move out of
the drawing board onto the
app store*



Contents

03 *Prologue*

04 | *Team Challenge*

07 | *Algorithmic Living*

15 | *App Trail*

27 | *Conclusion*

28 | *The Resume*

The Client Log

A Typical day at the firm

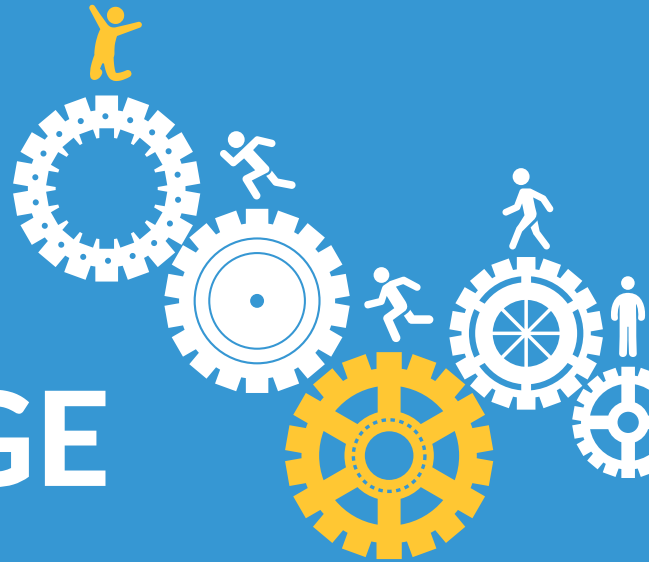
Production management is what happens on regular days. Running a business that offers development services, means you ship out products you make to someone else, who then owns and sells it to others. While it is hard to give up ownership, there are perks to being the most important cog in the wheel. A stream of interesting ideas will find their way to you. You will get to build some pretty awesome products or services that bring value to millions out there.

The usual day at the firm (for those who haven't guessed, we are a development services provider) begins with checks to see if production is on schedule. Resolving impediments or addressing requests for additional resources take up most of the work day.

Walk-in Days

But some days are refreshingly different. Days when prospective clients walk-in or reach out with interesting new ideas. If it is an idea that seems promising, you will want it to be built as much as the guy sitting across you. But, ideas need resources to materialise and patrons to survive. Both you and the prospective client will now embark on a shared quest to draw the perfect blueprint for the project. The customer has a rough blueprint and an initial set of specifications laid out for you. The specifications centre on themes like the choice of IT model, the business objectives, the product features and functionality. You offer a few suggestions based on your vast experience with product development. The client seems intrigued by some of the ideas put forth. He requests a detailed report that sets out the development process, the methodologies and infrastructure in detail. It's time to call for a team meeting.....

1 TEAM CHALLENGE



When a new idea finds its way to the firm, it calls for a team meet. We bring in the experts to solve the business and technology riddles presented by the client. What follows are lively discussions among team members, with ideas that garner most support making it to the whiteboard.

Leaning towards LEAN

The list on the board was impressive...



Google App
Engine



Lean Software
Development



Agile
Practices



Dev Opps

The ideas on there could all be grouped under the lean paradigm. Once the principles were agreed upon, the team settles down to chalking out a detailed strategy. The possible impediments and enablers are identified. Any learnings from previous projects are discussed. One final task remains. The contours of the project report are left to be finalized.

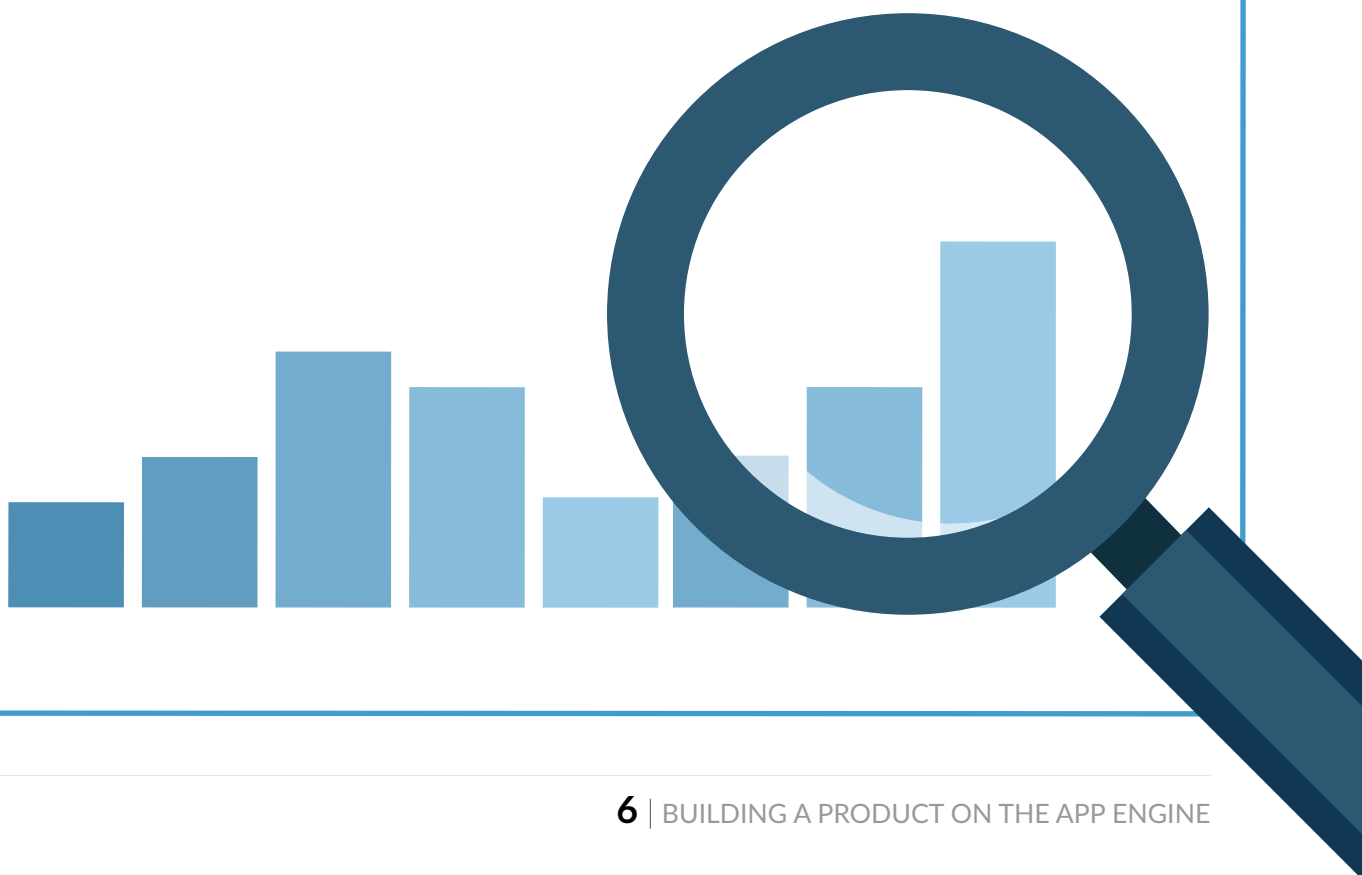
It is decided that the report will focus on the lean theme. The first part of the report will introduce the lean-agile paradigm. It will also convey the benefits that lean development practices will bring to the project. The second part of the report will outline the various stages of product development. The report also tries to highlight the complementarities between Google App Engine and the Lean paradigm where applicable.

Our pre-sales team is then assigned the task of coming up with a report along the decided lines.

The Report

The report is ready and is reviewed one final time. It does an excellent job of conveying the thought and process which the firm uses to create successful products.

It is now time for you to take a look....



2

ALGORITHMIC LIVING

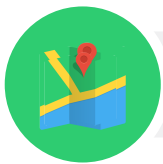
Code, the Language that runs modern life



We live in a bold world. Our world is constantly innovating smarter ways of doing things. Would software be left behind? Of course, not! Software is ubiquitous in modern life. If you pause to think, algorithms are the most important adjuncts to modern way of life. 'Code' helps you decode many things in life....for example:



Tinder: Helps you find your next date



Google Maps: Helps you get to your destination



Google Search: Finds answers to your queries



Amazon: Is your favourite shopping assistant

All of this and more is run by languages which we can only write but not speak. The process of writing algorithms/code is called development. The early models of software development were borrowed from those used in hardware industries. This crude adaptation of models from hardware industry worked well for large projects. For instance, the structured process of the waterfall model suits projects where requirements and scope are fixed, and technology is well understood. Conversely, it does not suit dynamic contexts, which favour a more evolutionary style of development.

Startups and large enterprises both want to avoid market failure. The lean movement is a product of this desire to minimise risk and maximise value.

How Lean is Not Traditional

The traditional business model is giving way to the lean start-up. The lean startup movement has a more exploratory approach to the business of launching a new service or product. A stark contrast to the old regime, which relied on the perfect plan, long-term forecasts and stealth working mode to get a product out into the market.

What 'Lean' People Know

The lean startup assumes that

- The business model is not perfect
- The right product-market fit will only reveal itself through trial-and-error methods
- Product success/failure can only be guessed at through real consumer feedback.

Lean methodology thrives on creating variation and learning from it. Different hypotheses are tested, revised and discarded. A continuous customer feedback loop is established both pre- and post- product launch. This strategy ensures that too much money isn't poured into ideas that would end up failing.

The Winning Philosophy

It's a sound philosophy. If you are competing in a game, it would be prudent to discover your weaknesses early on and work on improving them. The alternative would be to find out on field and suffer a crushing defeat with no hope of changing the verdict. If lean methods are widely adopted, they will reduce the incidence of start-up failure, which at the moment stands quite high.

The B Team

Alternative development methodologies like the Lean-Agile model and devOps movement are the B team. They have tried to address the limitations of the A team, a.k.a the waterfall model.

The newer lower risk models follow incremental, iterative processes that rely on continuous delivery and integration. The change in methodologies has been helped along by the emergence of cloud computing.

Given their popularity and apparent value, it is pertinent to introduce these concepts and understand the synergies between these methods and the cloud computing paradigm in general.

Lean-agile methodology



Think big, act small, fail fast; learn rapidly.



A new development hire in a large firm is usually given a lengthy manual of project documentation on his first week on the job. He may frown at the bulky manual, but once he pores over its contents, he is ready to join the team. The beauty of the traditional waterfall model lies in its simplicity and robust documentation. New members can easily replace old ones in such processes. This is absent in lean methods which often don't support extensive documentation. Despite this disadvantage, lean methods are rapidly gaining in popularity in the IT industry. The answer to this seeming contradiction lies in the 'business value' lean principles bring to firms, which also explains their popularity.

The Right Focus

Lean methodology owes its success to its laser-sharp focus on the right elements needed to ensure product success.

1. Build the right thing

Discover and deliver value to real customers

2. Build it fast

Reduce lead time. The time between customer order to solution delivery is reduced drastically.

3. Build the thing right

Build quality and integrity into the product with automated testing and integration

4. Learn through feedback

The product design must be perfected through frequent feedback right from the earliest stages of production.

LEAN principles

The lean-agile methodology has evolved its development practices based on a set of foundational principles. A cursory glance will help understand the specifics of the lean software development that would be introduced next.

1. Eliminate waste
2. Amplify learning
3. Decide as late as possible
4. Deliver as fast as possible
5. Empower the team
6. Build integrity in
7. See the whole

A New Manifesto for Software Development

Lean software development combines lean methodology with agile practices. The principles behind this method stem from lean manufacturing principles.

In the waterfall model, the software moves through set stages of design, development, testing and deployment. Testing and integration were usually left to the end of the development chain. The process meant that defects went undetected for a long time.

To counter the negatives of sequential processes, lean- agile software development practices focus on fast feedback cycles:

1. Test-driven development

Accumulation of defects is prevented by running tests as soon as the code is written. Automated testing is a part of this methodology and is used to build integrity into the product.

2. Continuous integration

The code is integrated into the code base frequently and in small increments. Simultaneous testing is done to check if the newly added code is running smoothly without introducing errors.

3. Iterative development

The software development cycle is kept short, with a working application being readied in short iterations of 2-4 weeks. The iteration is followed by review.

4. Cross-functional teams

If one goes by Lean-Agile believers, cross-functional teams are the best teams when it comes to software development. What is a cross functional team?

A team with members who have all the necessary functional and domain skills to successfully develop and deploy a software product. Such teams include developers, testers, project managers, systems administrators, release managers, and business domain experts among others.

Lean IT model (cloud computing)

Cloud computing IT model has replaced traditional model of IT provisioning. The cloud rents computing muscle along with range of management services. These have lowered the price and complexity inherent in software development greatly.

How the 'Cloud' is Lean

A. PaaS: Cuts down the development costs. Ensures critical developer time is not lost in routine maintainan

B. IaaS: Pay-as-you-use model. Ensures you pay only for what you use at each stage in the product lifecycle.

C. Infinite scalability: The cloud offers infinite scalability at competitive prices. In traditional models, provisioning for demands spikes meant that servers were kept idle for most of the year.

Cloud and Lean: The Recipe for Product Success

Cloud computing provides easy and automated provisioning of environments. The cloud is not just computing infrastructure (VMs and Network Resources). Cloud offers a range of partial to fully managed services like storage, analytics, PaaS and management services atop this infrastructure. Cloud is also pushing for newer architectures like microservices architecture which will underlie the next generation of cloud applications. The cloud framework is entirely congruent with the lean methodology principles. Let's understand why...

A. Eliminating waste through A/B testing

Google App Engine allows testing of multiple versions of the application. A/B testing and tracking cookies can help gauge consumer response to different versions of the application. This can help trim unnecessary features.

B. Build Integrity In through Testing

Google App Engine allows unit testing and integration testing of the code while it is in development. Identifying mistakes early saves time and money. Also, it allows quality to be built into the application early on.

C. Deliver as fast as possible through PaaS

Google App Engine is fully managed and hence applications can be developed and deployed in a short time. Reducing lead time is a stated goal of the lean model.

The Future: Lean as the Norm

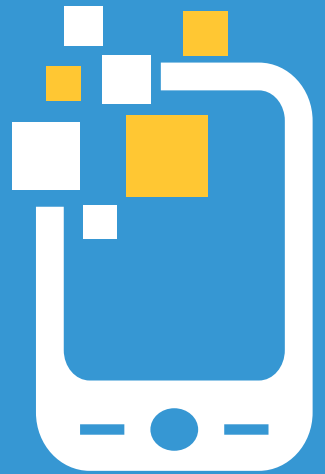
Lean start-up methods aren't just for young tech ventures. Large companies, such as GE and Intuit, have begun to implement them too.

Lean-agile methodologies will become the de facto software development methods in the next decade. Early adoption of these practices will increase product competitiveness and help businesses stay ahead of the pack.

3

APP TRAIL

How ideas move out of the drawing board onto the app store



The App trail is something everyone embarking on product development must be familiar with. If you are an enterprise CIO overseeing numerous in-house development teams you will be familiar with most of what we are about to tell you. Alternatively, if you are a startup entrepreneur with limited product development expertise you are in the best chapter of your life, so to speak. Why you ask?

Connecting the Dots: New Needs Breed New Methods

We have already introduced you to changes sweeping through the world of software development – the agile-lean methodologies, devOps and Cloud computing. These changes reflect the changes in IT environment as a whole. Cloud computing has democratized cutting-edge technologies that allow applications to be launched quickly and cheaply. This has sparked off an app race. The market is witnessing new and interesting use cases that debut new services or service old problems in new ways. Most of these new products are launched by startups headed by first generation entrepreneurs. The traditional heavy methodologies cannot serve this new clientele. So what software development model matches startup needs?

Agile-Lean Model: Development that sparks Discoveries

The startup clientele is a mixed package, a combination of creative abundance on one side and resource crunch on the other. Also, a startup idea is usually hazy at least on some of the details. Understandably so, given that founders are not always experts in the particular domain or market. In such cases, the optimum product-market fit is not clear at the outset and is usually achieved after a few iterations. The development process has to aid this discovery. Agile-lean, devOps and cloud computing paradigms are best suited to such dynamic development contexts.

Where it begins

It begins with the quest to build the 'Right Product'



One that finds early
market adoption



One that
delights customers



One that goes
Viral



One that is
sticky

The Premise



People don't know what they want until you show it to them.
-Steve Jobs



Before you give wings to your startup dreams, test your idea. Product validation through market analysis and concept testing is crucial. This is because even the most brilliant ideas are not immune to the vagaries of the market.

Solving the Outsourcing Dilemma: End-to-End Product Development

Does it mean cost concerns weigh higher than quality for most enterprises and startups? It's never that simple. True, outsourcing is a significant cost and time saver. But, attracting the best IT talent with all the different skill sets required for end-to-end product development is not easy or necessary. In a world where technology and skills become obsolete quickly, it is not prudent to spend time building teams whose skills may not serve you for long. It is easier to hire a firm providing end-to-end product development services. This is why OPD will continue to shine.

Why end-to-end product development?

An integrated design and development model where the development partner is brought onboard early on helps in clear understanding of the business objectives as well as user goals. This solves the quality issues and communication headaches inherent in outsourcing model.

Thank You for previewing this eBook

You can read the full version of this eBook in different formats:

- HTML (Free /Available to everyone)
- PDF / TXT (Available to V.I.P. members. Free Standard members can access up to 5 PDF/TXT eBooks per month each month)
- Epub & Mobipocket (Exclusive to V.I.P. members)

To download this full book, simply select the format you desire below

