**Week 5: Implementation Plan, Strategic Controls, and Contingency Plan Analysis**

Nate Bachmeier  
STR 581: Strategic Planning and Implementation  
University of Phoenix  
March 6, 2017

# Implementation Plan, Strategic Controls, and Contingency Plan Analysis

Renaissance Learning is an educational technology firm, which provides assessments for kindergarten through twelfth grade students. Based on the results of those assessments, teacher receive reports and other analytical tools for identifying gaps in the student’s understanding. Their software is served through a web site which is hosted in a private data center. In order reduce costs and increase development agility, they need an implementation plan for moving to a public cloud provider.

# Objectives

The primary objective of this work is to reduce costs associated with development and production workloads. While the physical servers in the private data center have long since been paid for, they require full time staff to maintain them. Each staff member costs the company $150,000 annually between salary, benefits, and required resources. If the number of employees can be reduced this will have a direct correlation to the profit of the business.

Another area related to cost is the speed of provisioning new infrastructure for the different workloads. Presently each new environment requires weeks of capacity and budgetary planning. However, once the business runs on the public cloud infrastructure, provisioning of those same resources is instantaneous. This reduces the time required for engineers to create new features and have them ready for customers.

A third objective is to update the legacy systems to use modern standards and practices. This will simplify hiring processes as these are the skills most prevalent in the current job market. Then through the laws of supply and demand the company can negotiate better labor contracts, also reducing costs. In addition, the conformance to new standards will improve the quality of the new cloud system, as it could then gain access to newer technology solutions.

The final objective is that the project can be completed before the start of the school year in August. Once school is in session the business tries not to touch the production site.

# Functional tactics

## Human Resource Management

The company lacks the expertise required to build solutions on modern technologies, and has an educational gap. While it would be possible to fill that gap, that would take too long, so new staff needs to be hired. There are other parts of the project that will need to be outsourced and overseeing those human resources will be required (Casio, 2014).

## Legal

As the business adds dependencies on various service providers, new employees, and outsourced labor; there needs to be assurances of quality and execution. Consider the scenario of hiring a business that does not even deliver the work. In these scenarios contracts need to be drafted to ensure responsible parties act responsible (Mallor, Barnes, Langvardt, Prenkert, & McCrory, 2014). Legal services will also be required to ensure that the business meets any regulation associated with storing student information.

## Marketing

The new cloud based solution will gain additional features, such as faster reports and access to larger data sets. The marketing team will need to give thought toward product, placement, price and promotion – to create a campaign to make sure customers are aware of these new capabilities (Kolter & Keller, 2014). There will also need to be coordination with the sales teams, so that additional licensing fees can be collected from those new features.

## Finance

Taking on a large project requires efficient planning of capital through budget management strategies. This project would be no different, as it will require significant engineering resources. Aside from the influx of labor costs, there will also be fewer people available to work on the legacy system.

## Operations and Supply Chain

The business previously used an ad hoc supply chain and operational model for building their software and serving it to the customers. This was possible as they had a finite amount of hardware, which was deployed in a static configuration. However, in the cloud these processes and pipelines need to evolve.

For example; the current supply chain has engineers writing code, next manual testers verify it for completeness, and finally operations team will deploy it. Using modern systems these steps are merged, such as engineers write code; then automated systems run tests and deploy to the environment. This leads to faster iterations while using fewer staff members. The additional staff can be reassigned or consolidated, allowing the company to do more with less.

# Action Items

## Configure Deployment System

For the new system to be successful, it needs to allow feature teams the ability to deploy in a repeatable manner. There are numerous technology solutions for automating this process, once needs to be chosen and provisioned. Once this layer exists templates and documentation needs to be created, so that feature teams can plug in their projects.

## Remove on Premise Dependencies

There are several systems within the web application which use corporate resources, and those resources will not be moved into the cloud. For instance, the user management service uses a central Microsoft SQL Server 2008 instance. Once the software moves to the cloud, there will not be a concept of a central shared database. This requires that software component owners to identify these dependencies and remove them.

## Integrate Disjoined Systems

The company has roughly 120 engineers which are working across fifteen feature teams. Each team is assigned a different area of the release and working in silo. There needs to be a strategy which the different silos can converge and form the end to end customer experience.

To accomplish this, there must be well defined data and service contracts so that each component knows what they will be sending and receiving. The architecture team also needs to have well documented patterns so that communication between the components is consistent.

# Milestones and deadlines

## Alpha Release

This engineering effort needs to start in January, and be ready demo individual components working by March. At this stage, there will be simulated traffic and the system will end to end experience will not be operational. The deployment solution also needs to be available at by the end of this milestone.

## Beta Release

Between March and July there needs to be an integration effort which gets the systems to communicate with each other. A lot of time and energy needs to be invested into making the system highly reliable and production ready.

## RTW Release

Finally, between July and beginning of August, the system needs to be provisioned into the production environment. The quality bar needs to be very high and there can be few changes permitted into the system. Otherwise there is regression risk and customers may experience outages.

# Tasks and Task Ownership

Each feature team will need to implement the action items listed above, this is because they are the experts in their respective parts of the code. By distributing the work amongst everyone it also increases the chances that the effort is finished on time.

There will also need to be some level of cross communication between feature teams, so that work can be negotiated and dependencies come together correctly. These tasks need to be identified, and a team lead should be responsible for overseeing those requirements.

The operations team will be responsible for provisioning the deployment system, and setting up the billing requirements with the cloud provider. After they complete that work, it will be the responsibility of each team to integrate their changes into the operation team’s environment.

# Resource Allocation

For each of the feature teams to be successful, they need to have a complete team with engineers, operations, project managers, quality assurance, and leaders. If there are insufficient members on a team, then that team will have challenges completing large projects.

Those teams must also have access to software resources needed to perform their job. These would include access to cloud resources, software editing tools, physical laptops, an office building to work from, and Internet access to name a few.

# Recommend any organizational change management strategies that may enhance successful implementation.

One of the challenges of the organization is that it leverages a matrix topology, where multiple leaders share the same engineering resources. Under a matrix organization there is a potential benefit that there is more communication among all employees (Robbins & Judge, 2014). However, this adds scheduling contention as many people want many things from the same resource.

An alternative approach would be a functional organization, which has one manager for each employee. When there is clear reporting of employees then their priorities can be easily set and managed. There is also more direct communication as there is only one channel to listen, instead of many.

# Develop key success factors, budget, and forecasted financials, including a break-even chart.

The project will be successful provided it costs less than the current system, and it can be completed on time. There are many factors which come to into play when evaluating the future forecasts. For instance, how many staff will be repurposed versus removed? Or how many redundant deployments are needed to have a fault tolerant system? Or how efficiently can components communicate together, in practice?

Maintaining the production environment will cost roughly 250,000 less per year after moving to the cloud. This would make the break even on maintenance occur after six month to one year. The cost of software engineering is not included in the analysis as they are a sunk cost, which excludes them from analysis (Kimmel, 2014).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Private Datacenter | Item | Quantity | Cost | Cost/Year |
| Servers | 10 | 15,000 | 50000 |
| Hardware Engineer | 3 | 60,000 | 180000 |
| Networking Engineer | 3 | 75,000 | 225000 |
| Software Licensing | 1 | 25,000 | 25000 |
| Support Contract | 1 | 100,000 | 100000 |
|  |  |  |  | **$580,000** |
|  |  |  |  |  |
| Cloud Solution | Compute Hours | 36500 | 1.5 | 54750 |
| Network Gigabytes | 10000 | 0.25 | 2500 |
| Storage Terabytes | 15 | 240 | 3600 |
| Hardware Engineer | 2 | 60,000 | 120000 |
| Networking Engineer | 2 | 75,000 | 150000 |
|  |  |  |  | **$330,850** |

# Create a risk management plan including contingency plans for the identified risks.

The biggest risk to the business is a failure to deliver the new system on time or with high enough quality. This would cause the release to be delayed until the next school year. By that time, competitors might catch up and seize the opportunity. To mitigate these risks the business needs to realistic estimates of time required to complete each task. There also needs to be a clear understanding of which asks are must, should, or nice to have (Robbins & Coulter, Management, 2014). That would enable the business to cut additional features while still delivering a complete solution on time.

The second risk to the business would be that the new system costs more than the old system. Public cloud solutions are based on micro transactions, and can easily cost thousands of dollars more than initial estimates. This is especially the case during the development phase, where numerous additional resources are provisioned to handle different builds of the product.

# Conclusion

The business will be able to save money by moving their operations to the cloud, instead of relying on a private data center. However, the implementation of this strategy will touch every area of the business. Thought needs to be given to ensure the delivery of a unified product delivery across the different functional units.

There are also several high-level action items that need to be broken down into tasks for the feature teams to perform. As they perform the tasks they need to be causious of where in the schedule they are. If they exceed any time limits, there is risk to the entire corporate strategy. To mitigate those scheduling risks, leadership teams need to have a list of items they are willing to cut.

The organization might also find value in revising their organization structure so that there are more clear lines of communication. Though none of that will matter unless the project is completed on time and on budget. It appears that the project will start to save money within the first year and make the company even more competitive going forward.

# References

Casio, W. (2014). *Managing Human Resources: Productivity, Quality of Work.*

Hitt, M., Ireland, D., & Hoskisson, R. (2014). *Strategic Management: Concepts and Case Studies.*

Kimmel, P. (2014). *Accounting: Tools for Business Decision Making.*

Kolter, P., & Keller, K. (2014). *Marketing Management.*

Mallor, J., Barnes, J., Langvardt, A., Prenkert, J., & McCrory, M. (2014). *Business Law.*

Robbins, S., & Coulter, M. (2014). *Management.*

Robbins, S., & Judge, T. (2014). *Organizational Behavior.*

Ross, S., Westerfield, R., Jaffe, J., & Jordan, B. (2014). *Corporate Finance.*