Planning and executing fixed bid projects using Agile

Vijay Iyer
Delivery Principal
ThoughtWorks Inc.
I've been at ThoughtWorks for about 13 years.

I've led large (often globally distributed) teams of consultants building application software for our clients using agile, lean and XP methodologies, hitherto always on a time and material contract basis.

This was my first time executing a delivery of this scale as a fixed bid project.
### The fundamental disconnect between fixed bid contracts and agile delivery models

<table>
<thead>
<tr>
<th>CONTRACT NEGOTIATION</th>
<th>FOLLOWING A PLAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVER CUSTOMER COLLABORATION</td>
<td>OVER RESPONDING TO CHANGE</td>
</tr>
</tbody>
</table>

Fixed bids transfer the financial risk of delays to the delivery team. This results in both sides focusing on iron clad contracts to protect their interests. Collaboration with the Product Owner necessary to develop the best product within the given constraints takes a lower priority.

To enter into iron clad contracts, the Product Owner is forced to lock down scope very early in the project lifecycle. The lack of flexibility during the development iterations leads to a less optimal solution for the business and customers.

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan
The initial plan was for ThoughtWorks to build the middleware and the data access layers of the new platform, and for the internal development team to build the UI components.

While the ThoughtWorks team had completed multiple releases of the middleware and the data access layers of the new platform, the customer/agent facing portal was way behind schedule.

ThoughtWorks took over the delivery for the customer/agent facing consumer loan portal on a fixed bid contract.

The consumer loan product was launched with full servicing through the customer and agent portal.
The obstacles on our road to success

- A very tight timeline and lot of scope to cover
- Lack of trust from disillusioned and frustrated product owners
- Pressures from Finance and Legal teams on both sides to “fully define and document” requirements and technical design
- Unsupportive and distrustful client development teams
- No sense of ownership or urgency within the teams that we had dependencies on
- A ton of existing documentation
After each milestone, we reviewed progress and altered course accordingly.

Discovered
We kicked off the engagement with a 2 week inception.

Planned
Using what we learned, we defined the release plan and the terms of the engagement.

Developed
Built the application based on the release plan identified.

Released
At every milestone, we released into production.

Reviewed
After each milestone, we reviewed progress and altered course accordingly.

Our Execution Model
Benefits of the project inception

An opportunity to regain trust and build a relationship with the business stakeholders

Validation of state of existing code helped build better estimates

Definition of deadlines and working relationships with dependency owners

Clear agreement on the “must have” scope for the customer and agent portal
Benefits of a flexible SOW

Commitment to development effort helped maximize business value for the client, while reducing the delivery risk to ourselves.

Product owners also received the flexibility and the lead time to spell out complete requirements closer to when they are actually being worked on.

The exit clause at the end of each milestone gave each party checkpoints to take corrective actions.

Scope Baseline
Scope baseline has been achieved through a combination of information available in the existing project requirement repository, including limited screenshots and wireframes, without an opportunity to engage a broader base of Project Owners and/or functional or business stakeholders to elaborate and define at a level of granularity beyond what has already been specified by Project Owners - that ideally allows for a fixed fee arrangement. Base-lined scope in the form of an Initial Story List (ISL), with points assigned for each story during the Requirements Elaboration/Inception, and low fidelity wireframes are attached in the Appendix A - also listed are a set of Acceptance Criteria that forms the basis for accepted behavior and user interaction and, ultimately, acceptance of the Milestones.

Change Control
- Any major deviation to defined scope in ISL that impacts design or functionality will need to be approved by the program's Change Control Board, within the constraints of the project budget.
- The Change Control Board can choose to manage scope by swapping agreed to functionality to cater to new (higher priority) functionality of equal size to come into the backlog. If there is no existing scope which can be swapped out of the backlog and the new functionality also is business critical (for launch), the parties will mutually agree to adjust the cost of the project based upon ThoughtWorks' estimates for the additional work to be done on Channels.
Benefits of milestone based planning

Payments and penalties were linked to the timely review and sign off of milestones, encouraging joint ownership.

The various integration points with other systems, external feeds etc. could be tested and fixed much earlier in the cycle.

The frequent milestones gave an opportunity to repeat, refine and streamline the release process.
Benefits of parallelization

Parallelization enabled the team to start work on different sections at the same time early in life of the project.

This gave the product owners early visibility into and feedback from the half-finished product.

This allowed us to change some of the original ideas around the look and feel and navigation behavior.
Benefits of focusing on reusability

By building the foundational components first, we could reuse them efficiently to layer on additional requirements and changes.

This reduced development efforts for the later components to a matter of days or even hours.
The milestone plan exposed within the first 6 weeks, the complete lack of readiness of the production deployment teams at the client.

The early progress and responsiveness on feedback is something the Product Owners hadn’t seen from their own teams, and helped build a lot of trust and goodwill for the ThoughtWorks team.

We not only finished the planned milestone deliverables ahead of schedule, but also managed to bring in several additional nice to have features, greatly delighting the Product Owners.

The project was highly successful commercially, but at the same time, the team also enjoyed the engagement very much.
Learnings from the engagement

**Engagements should be shorter term (3-6 months) in duration, and linked to a specific business goal**

The volatility in the project environment increases as the timeline increases.

There is a clearer sense of urgency and commitment around a specific business goal with explicit business value.

**All parties in the critical path should have the same sense of urgency and same level of commitment to project goals**

Calling out explicit commitments from dependencies like external vendors, infrastructure, security ensures they play a more collaborative role and give timely attention to the project.

**Fixed bids get exponentially more complex with globally distributed teams**

Virtual collaboration tools greatly help to keep locations in sync.

A small onsite team (business analyst and senior developer) purely for communication and blocker elimination also makes things easier.

The team doing the work estimates!

**Post inception, a rigorous cadence is necessary to ensure adherence to the plan**

UX design should be done prior to the start of the milestone using lo-fi sketching.

Story narrative writing and technical analysis should be done at least two weeks ahead of development to ensure a smooth story flow.
My team

A big thank you to my team without whom this project wouldn’t have been so successful!
THANK YOU

CONTACT

Vijay Iyer
vijay..iyer@thoughtworks.com

ThoughtWorks Inc.
99 Madison Ave
New York, NY 10016
USA