The Value of Local Developers

A comparison of local and outsourced development processes in Higher Education

Commissioned and reviewed by: **Mahendra Mahey** for the JISC funded Developer Community Supporting Innovation project (DevCSI) based at UKOLN, University of Bath

Authors:
Malcolm Ramsay
Mark MacGillivray

A Cottage Labs report

Table of Contents

1. Executive summary

In this first phase of research we analyzed the Cambridge Library Web Services (CLWS) software development project. The initial findings point to some significant differences in user engagement between local and outsourced development projects.

Through conversation with the development team potential outline metrics were identified and quantitative estimates provided to describe stakeholder engagement and timescales.

Initial estimates suggest the Cambridge Libraries Web Services development project achieved 100% stakeholder consultation and high user engagement rates of 29%, compared with less than 1% engagement for a comparable outsourced project.

2. Introduction

The DevCSI (Developer Community Supporting Innovation) project (http://devcsi.ukoln.ac.uk) has been driven by a number of assumptions about developers in Further and Higher Education, their role in supporting innovation and the benefits that may bring.

An initial DevCSI stakeholder survey exercise in 2010 was carried out which aimed to:

- test and provide some evidence to support some of these key assumptions
- ask stakeholders to provide example stories of the value of impact of what developers did, these would be explored in greater detail later
- help shape the future of DevCSI activity

This report has been commissioned to initiate identification of metrics that enable a quantitative description of the impact and value of the work that local developers bring to their institutions.

This work builds on initial findings from the 2010 DevCSI developer survey, and drills down on a specific case study identified by Andrew McGregor and Mahendra Mahey, as 'representative' of the community that the DevCSI project serves - namely 'local developers' working within Further and Higher Education.

This preliminary analysis provides a basis for a more detailed discussion of metrics related to the measurement of value chains within the software development ecosystem in Further and Higher Education.

2.1. Case Study Details

Based on responses to the DevCSI Stakeholder survey, DevCSI selected the Cambridge University Library reporting project as a suitable case study for investigation. This project was a local development designed to improve library reporting and library management systems at the University.

A qualitative report was previously commissioned (Author: Michelle Pauli) to describe the development project.
Key quotes from this report include

> *“The University of Cambridge libraries team was using a commercial library management system that could not cope with the demands of specialised reporting for the 70+ libraries it supported. Huw Jones, a systems support librarian with developer experience, had been brought into the libraries@cambridge team from a college library on a part-time basis. He quickly identified the issue and spotted the way out. “*
>
> *"We were locked in a cycle of spending all our time running to keep up and not having time to look around and see what could be done better. It was an investment in time to get time back," says Jones.*
>
> *“It has saved the IT staff time, as they no longer have to print out stock checks and walk over to librarians with them, and the new system can identify and correct cataloguing errors automatically.”*
>
> *“A total of 57,000 reports across all the libraries were produced last year, of which 36,500 were scheduled reports, including daily circulation stats which were previously run monthly with far less detail.”*

This case study is available via the DevCSI blog, http://devcsi.ukoln.ac.uk/blog/local-developer-impact/

2.2. Analysis of assumptions

In order to proceed with an outline of suitable metrics we will consider two main scenarios, namely, local development (i.e. development carried out by developers working within the organisation) vs custom outsourcing within an educational context. Below, we describe several assumptions relevant to these scenarios which this report will attempt to address:

- Assumption 1 - A key difference in local development projects is the level of contact and feedback between users and the development team.
- Assumption 2 – A greater amount of contact translates into a faster build time and many more iterations of the solution as feedback can be almost instantaneous.
- Assumption 3 – Resulting solutions based on assumptions 1 and 2 are better suited to users’ requirements. As a result there is higher user satisfaction with the finished tool. This will also translate into greater engagement/ usage of the product.

- Assumption 4 - local developers may be more expensive on a crude, initial cost analysis from circumstantial evidence but over the longer term there are likely to be cost savings.

- Assumption 5 – One of the benefits of local developers is that there is a general blurring of the development process. As a result there need be no clear cut end point to the software release cycle. Therefore there is less emphasis on creating a deliverable at specific points of the process, and more emphasis on delivering working code that meets user needs, resenting an ideal environment for agile software development

- Assumption 6: The quality of interaction with a local developer and internal users is likely to be higher than with an external outsourced company, as it is informal, relaxed and based on familiarity. Knowledge transfer effort should therefore be minimized.

Based on these assumptions, metrics were considered that could capture the key aspects of developmental work from a Further and Higher Education perspective. This should include but not be limited to:

- cost
- timeliness
- resources
- capital expenditure
- end user satisfaction
- fitness for task
- open operability of solutions
- ongoing license fees
- ongoing operating costs

2.3. Interviewees

As part of the initial data gathering exercise we interviewed key personnel to understand the problem space in more depth.

Through this process we spoke to:

Lesley Gray - Administrator & Library Co-ordinator
Huw Jones - Lead Developer, Assistant Library Officer
Edmund Chamberlain - Systems Librarian

3. Alternatives

While the Cambridge Libraries Web Services project was developed using in house developers, interviewees made mention of other projects using other resources. While each development project is different there are several broad categories of outsourcing to consider to ensure that like for like is compared. In pursuit of completeness we here briefly outline the main key alternatives:

- **Outsourced custom tools** — This covers scenarios where an external company is responsible for developing a novel software solution to match the needs of the F/HE institution.

- **Outsourced off-the-shelf tools** — This is largely similar to the above except the external company merely reworks an existing software solution to match the needs of the F/HE institution.

- **Mixed Local/ Outsourced** – This is perhaps the most common scenario whereby the F/HE institution relies on a mix of software both custom and off-the-shelf written by both local and external developers.

- **Outsourced IT** – this differs from the above in that the task of operating the tools is also carried out external to the F/HE institution

- **Business Process Outsourcing/ Knowledge Process Outsourcing (BPO/ KPO)** - in these scenarios admin and higher skill set functions are carried out by an external company. This can often replace the need to develop software to automate a task by instead handing repetitive administration to an external company

4. Data sources

Although Cambridge Libraries did not explicitly capture data to compare the development of this project with outsourced alternatives, they were able to provide values based on experiential estimates and comparable development projects.

Huw Jones, the lead developer on the library web services project, was able to provide input based on first hand involvement in the development process. As a developer working on site he was uniquely positioned to provide data as he led the process from the beginning. Specifically this gave insight into timescales and user interaction and engagement levels.

Additionally we were able to gather estimated values for an outsourced project based on previous project work carried out for the library. This was based on a comparable software development the library undertook that required a similar amount of development time. This project was designed to improve functionality of the Library’s resource discovery platform. The library purchased an off-the-shelf solution however there was still significant external development required to get the software to a stage that matched the library user’s needs.

In terms of people days of development time this was a similar length and had similar usability requirements. These initial estimates are based on the experience of staff working with external developers.

5. Metrics

In considering suitable metrics to describe the value of local developers it is useful to first iterate the problem we are looking to solve. This can be described as an attempt to quantify the difference in efficiency between local development teams and outsourced external developers.

Metrics should then aim to capture the differences between these alternative approaches to software development and measure the efficiency at various stages of the process.

This may include differentiation of:
- Software Release Cycle (prototype, alpha, beta etc)
- Scale (small group, larger departmental, institution wide, inter-organization, etc)
- Cost (<10k, <50k, <250k)

In choosing suitable metrics to capture relevant aspects of the process we wish to consider those indicators that are illustrative and also easy to capture. Given the fact that it is unlikely that most teams will have been explicitly measuring these factors during the development period it is also important that these metrics can be easily applied retrospectively.

The metrics developed here are suggested as a basis for future research, allowing departments a framework to build on in collecting and analyzing data from future projects. As such these metrics will be open to further refining as further projects are included.

It is also suggested that these metrics can form the basis of online tools allowing management to enter data about development processes and compare against aggregated ‘standards’ for the sector.

Below we list some potential metrics that apply to this situation:

5.1. User Engagement

- Total number of users/stakeholders
- Number of contacts between development team and users by phone per day
- Number of contacts between development team and users by email per day
- Number of contact points face to face
- Number of contact points (counted as a connection between any member of the development team and any user via any medium – phone, email, text, oral, etc)
- User satisfaction

5.2. Iterations

- Number of reiterations of requirements
- Number of releases produced

5.3. Costs

- Budget (£)
- Training costs (having developers onsite can effectively make training a subsumed ongoing process)
- Time (from project conception to initial prototype)
- Time (from project conception to alpha)

- Time (from project conception to beta)
- Time (from project conception to operational beta)
- Time (from project conception to full roll-out)
- Time (post roll-out)

6. Results

6.1. User Engagement

The key area of difference for local development versus outsourced development was in user engagement. Based on initial estimates there was a huge difference, as much as two orders of magnitude, between the level of user engagement in each scenario.

Table 1 shows the number of users participating to varying degrees in each project. Although there are some differences, notably in size of user base, the projects were comparable in terms of development time required and need for user input which were the factors under consideration

Table 1 – User Engagement

User Engagement	Local Development	Outsourced
Total user base	350	4000
Total stakeholders consulted	350	45
Stakeholders providing input	100	29
Regular contributors	25	8
Key Stakeholders driving project	13	2

Graph 1 - User Engagement

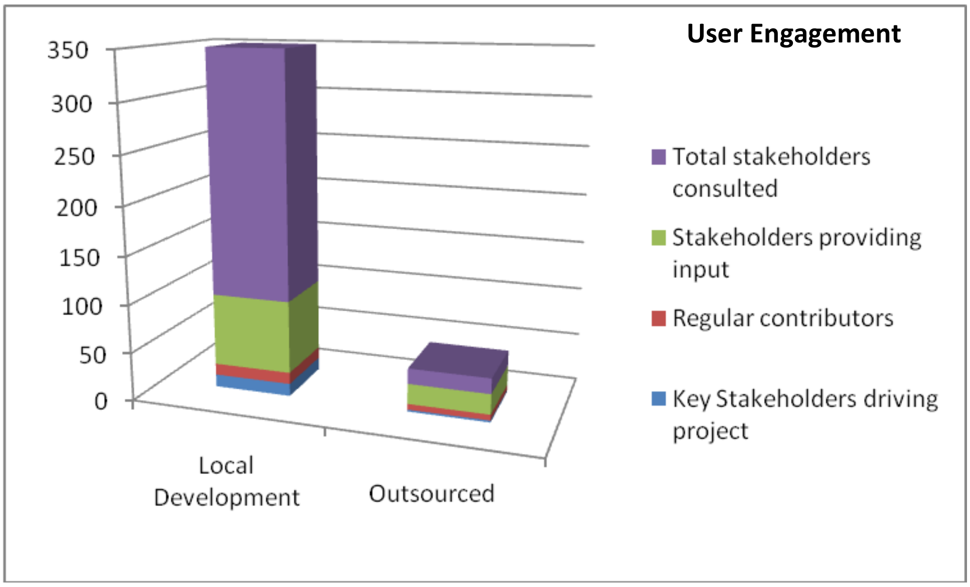
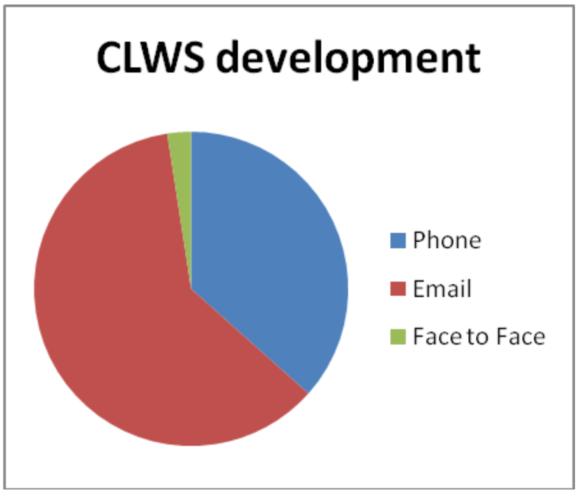


Table 2 shows user engagement as a percentage of total stakeholders and clearly demonstrates the difference in interaction between the two development processes. Where the local development project was able to consult 100% of users about their requirements only 1.1% were contacted in the outsourced project. More interestingly of those consulted 28.6% gave some input or engaged with the project in the local scenario compared with just 0.7% for outsourced.

Table 2 - User engagement as percentage of total stakeholders

User Engagement	Outsourced	Local Development
Total % stakeholders consulted for input on requirements	1.1%	100.0%
Stakeholders% s providing input to the development of the software	0.7%	28.6%
Regular contributors (providing input about user requirements)	0.2%	7.1%
Key Stakeholders driving project	0.1%	3.7%

Graph 2 - Contact frequency via different mediums



Contact frequency per day	
Phone	15
Email	25
Face to Face	1

- 10 -20 contacts by phone per day
- 20 -30 contacts by email per day
- 1 face to face meeting per day

6.2. User Satisfaction

Users of the system were reported to be 'very satisfied' with the resulting tools that were made available as the 'system was built to meet their needs'. Flexibility and quick turnaround time from suggestion to implementation were noted as key factors. **Implementation often took place on the same day for minor issues and this resulted in many users feeling it was "their" system.**

> No quantitative data was captured by Cambridge Libraries to measure this aspect of development (e.g. a survey) however it is suggested, based on qualitative input, that this should be a strong area for investigation in future research.

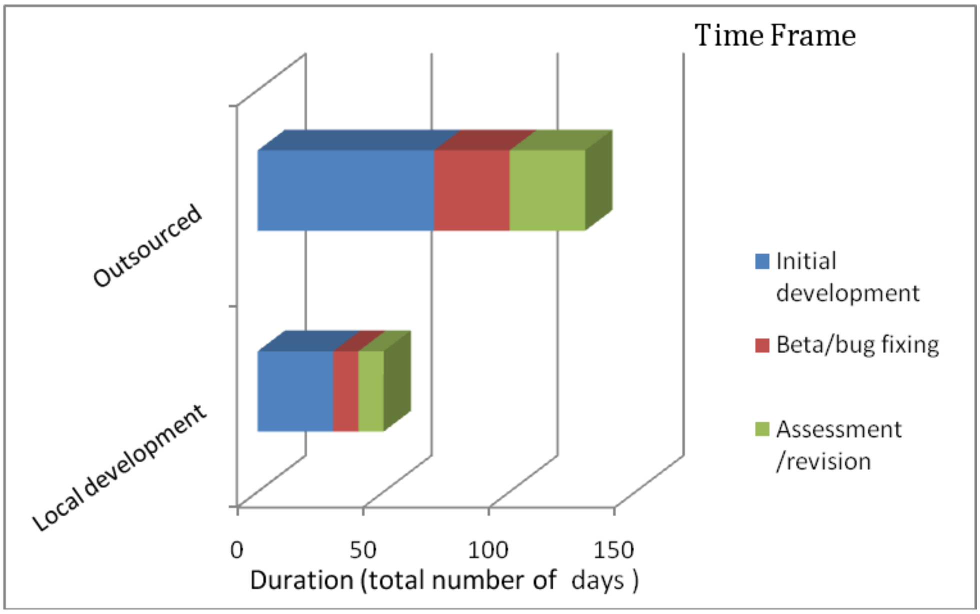
The concept of best practice is seen as an important part of future research and as such the level of user satisfaction is likely to be related to increased awareness and involvement with the process of collecting data.

6.3. Timescales

Table 3 - Comparison of time taken to complete local vs outsourced project

Comparison of time taken to complete local vs outsourced project	Number of days	
	Local development	Outsourced
Initial development	30	70
Beta/bug fixing	10	30
Assessment /revision	10	30

Graph 3 - Time frame taken for completion of local vs outsourced projects



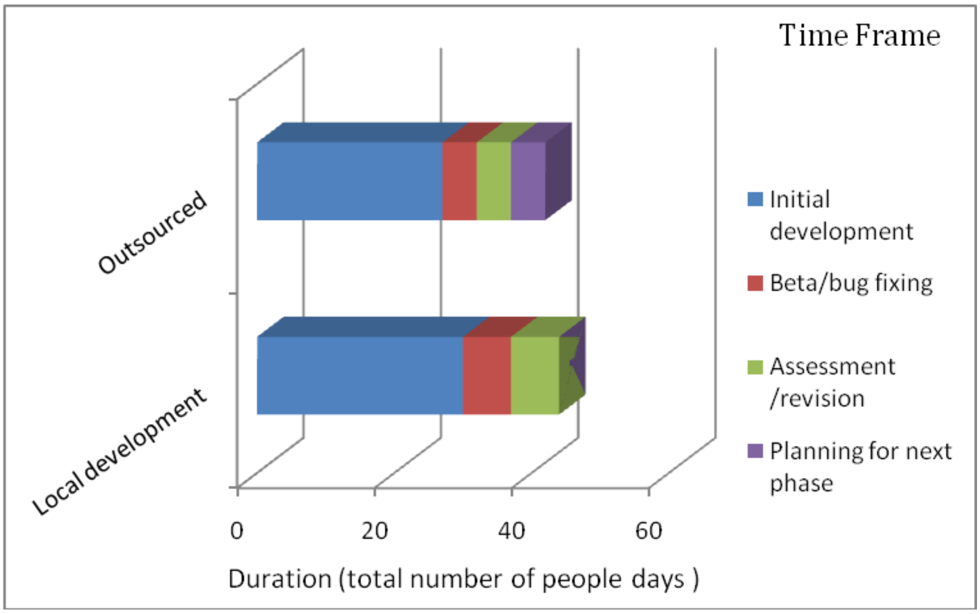
As table 3 shows, the outsourced project took considerably longer to reach completion in terms of total duration. While these numbers are estimates it is important to note that the actual number of people days involved is estimated to be comparable.

Table 4 shows the estimated people days involved in each project and highlights the efficiency benefits that the local development project achieved by having the work process embedded in the day to day work flow of the department rather than waiting on external milestones to be achieved.

Table 4 - Comparison of actual development days needed to complete local vs outsourced project

Comparison of time taken to complete local vs outsourced project	Number of people days	
	Local development	Outsourced
Initial development	30	27
Beta/bug fixing	7	5
Assessment /revision	7	5
Planning for next phase	0	5

Graph 4 -Number of people days required to complete local vs outsourced projects



6.4. Costs and time

Cambridge Libraries estimated the total cost for initial development including all modules and ongoing maintenance to be £30,000. This was mainly attributable to developer salaries and hardware costs.

Training took place informally within the team and semi-formally with training days organized for end users. As a result of this approach new librarians coming in tend now to be trained by their colleagues creating potential cost savings on comparative outsourced training.

No costs comparisons for overall training were available from Cambridge Library as much of it was completed informally by sharing within teams. However, a total of 3 formal training days were organized with an approximate salary cost for the trainer based on a yearly salary of £30,000. This translates to approximately £116 per day. As a rough comparison the approximate cost for an external training company in library software training might be around £800 per day.

This is therefore an area worth studying in more depth as the training process can present additional cost savings over the longer term by further boosting staff engagement.

7. Conclusions

The initial phase of this research points to significant differences in user engagement between local and outsourced development projects. Based on the preliminary figures for the Cambridge Libraries Web Services project, an estimated:

- 29% of all stakeholders were engaged with the project
  - of which 7% were regular contributors
  - and 4% were considered to vital in driving the project
- This compares with less than 1% total engagement for stakeholders in a comparable outsourced software project
- In terms of timescales, the outsourced project took 2.6 times longer took complete despite requiring less actual days. of development

8. Recommendations

Local development was reported to increase efficiency on several levels but in most cases quantitative data had not been collected. These phase one results are naturally crude, but provide clear, early indications of future research. This should include focussing metric definitions, conducting further in depth case studies, gathering more accurate data, and exposing benefits to the community.

It is recommended that further research should seek to:

- **Gather more data points** – there is a clear need to include a wider selection of case studies in order to provide more robust findings. Specifically to examine different aspects of the development process in more depth.
- **Increase granularity** – there is scope for greater granularity of metrics to better compare specific scenarios and define successful projects more accurately
- **Investigate user satisfaction and usage patterns in more depth –** there is a need to formulate an approach that takes into account and attempts to quantify more qualitative aspects of user satisfaction usage patterns. Particularly in order to quantify the amount users feel a system is ‘theirs’
- **Understand the role in mitigating training and support costs**– further analysis of the role that local development plays in mitigating ongoing training and support costs by distributing expertise within a department or group

- **Consideration of existing definitions -** it may be necessary to reconsider established software cycles and metrics due to the blurring effect associated with rapid prototyping in local development. Further research is required to examine how best to describe the process accurately and compare like with like between local and external development efforts.
- **Develop collection tools to increase awareness -** further research should provide quantitative measures and advice to community members facing similar situations. Research results should therefore be public exemplars that demonstrate how projects can self-analyse and share their own development stories. This process should be supported via simple online analysis and collection tools.

Cambridge Libraries were able to suggest opportunities for further investigation based on existing scenarios where local and outsourced developments have been conducted in a similar environment. Most notably contact was suggested with the University of Sussex Library and in particular Chris Keene, Technical Development Manager. Beyond this the most robust findings are likely to be developed based on a more real time approach whereby analysis could be made of two projects (local / outsourced) as the projects are underway.

Appendix A

Notes: Interview, Huw Jones

Contact

- number of contacts between development team and users by phone per day. Approximately how many times did you speak to any user per day by phone about the development?

  Internally, within team, constant contact (in same office!), realistically 5 conversations relating to development per day

  End users – constant contact by phone via helpdesk informed development by giving us a sense of user needs and expectations. 10-20 calls per day

- number of contacts between development team and users by email per day. Likewise for email

  See above – c. 20-30 emails per day (initially, far fewer as system was rolled out!)

- number of contact points face to face? And in person

  Externally – via library visits and training sessions, c. 5 times/week

- number of contact points (counted as a connection between any member of the development team and any user via any medium – phone, email, text, oral, etc). You mentioned ~350 in total- maybe we could break this down into -

  - Total number of contacts – 350 users/5 team members
  - respondents who provided some input – 100 users, all team members
  - respondents who were in regular contact – 25 users in regular contact
  - key users that helped to drive the development – 10-15

- Satisfaction? - how pleased are users when using locally developed software vs outsource. Is there any evidence of a link between involvement in the development stage and satisfaction/engagement with end product?

  - Number of people using the system now vs previously?

  System is now fully integrated into main workflows – all staff use it

  - Subjective user satisfaction

  Very satisfied – system was built to meet their needs. Extremely happy with flexibility and quick time from suggestion to implementation (often on same day for minor issues). Feel like it’s “their” system.

Time / iterations

- Do you have a measure of the time taken to complete various stages?
    - From conception of project to first prototype? – 3 months, v. part-time (2 days/week?)
    - From first prototype to first user testing? Was this instantaneous? Could users try the tool as soon as you wrote new code? Yes, we rolled out as soon as we had something that worked, and rolled out fixes and minor improvements pretty much "live"
    - From conception of the project to a production ready version that was used on a day to day basis?
      4 months to steady day-to-day use
- number of reiterations of requirements. How many times did the requirements for the tool change? Twice per day? Once a week? Not at all? How does this compare with a commercial solution
  At first, pretty much constant revision for the reporting tool (once per day?). Then, as system stabilised, able to move to larger revisions and extensions. Not possible with commercial solution
- number of versions released – was there a formal release schedule? Is there any way you quantified the various stages of development

We released in "modules" – reporting, stock-check, course booking, staff management, library directory, bibliographic enrichment and error checking etc.

Model for each module was – development (1-2 months), early beta release and intense bug fixing (1-2 weeks), assessment and (occasionally) more major revision (1-2 weeks max), reassessment of general needs and planning for next module.

Basically, initially we isolated things which were a pain in the neck for us or for users, and made them easier. Later, building on this platform, we were able to provide whole new bits of functionality.

Costs

- Do you have a value you can provide for the overall Budget (£)

Ed has estimated c. 30,000 for initial development (all modules), plus ongoing maintenance

- training? Did having a developer onsite effectively make training an ongoing process? Is there any ballpark figure you could provide on savings in training costs? i.e. you could train people through informal discussion rather than formal 'training days' with an external consultancy?
  Training took place informally within the team and semi-formally (i.e. we ran some training days) for end users. Now new librarians coming in tend to be trained by their colleagues.