Chang, M. et al. (Eds.) (2019). Proceedings of the 27th International Conference on Computers in Education. Taiwan: Asia-Pacific Society for Computers in Education

Diagnostic Language Assessment: Lessons Learned from Rapid Prototyping

Allan NICHOLAS & John BLAKE
Center for Language Research, University of Aizu, Japan
anich@u-aizu.ac.jp

Abstract: In this paper, we provide the rationale for a computerized diagnostic language assessment (DLA) of email writing for Japanese undergraduates. The development of a rapid prototype of the DLA system is then described, and lessons learned from the process are shared. The DLA focuses on *pragmatic competence* – the ability to alter one's language choices relating to politeness, formality and structure, depending on the social context of the communication. While tools now exist to assist with learners' grammar problems and vocabulary, few exist for this pragmatic aspect of communication. Furthermore, traditional language assessments typically provide an overall score of learner ability without identifying specific strengths and weaknesses. The DLA seeks to address this issue via a three-stage process: 1) assessment administration; 2) provision of feedback to the learner, and 3) provision of further individualized instruction based on the assessment results. To gain a better perspective of the problems and to start to climb the learning curve as quickly as possible, rapid prototyping using Axure RP was selected. This high-fidelity prototype of the DLA was designed for some use cases. Numerous lessons were learned. Notably, rapid prototyping is not necessarily rapid. Early feedback on usability and the user experience were received from the rapid prototyping, helping the developer gain a better understanding of the user needs and discover any unexpected usability issues prior to actual coding. Another ongoing advantage is that rather than describing the expected functionalities of the software to stakeholders or interested parties, the functionalities of the fully-fledged DLA can be shown using the prototype. The prototype functions as an interactive visual aid, reducing miscommunication and providing a focus around which discussions can be based.

Keywords: EFL, diagnostic language assessment, pragmatics, rapid prototyping, Azure RP

1. Introduction

As part of an ongoing larger project, in this paper we describe and reflect on the development of a rapid prototype of a computerized diagnostic language assessment (DLA) for Japanese EFL learners' English email writing. The particular focus for the DLA is on the pragmatic aspect of learners' email writing – the relationship between the social contexts of the emails and the language choices the learners make. We put forward a rationale for the use of DLA in the EFL classroom, and the ways in which computerization can enhance the potential of this form of assessment. We then describe the steps taken to develop the basic DLA prototype, and conclude with the lessons learned by the developers from the prototype development process.

2. Background

2.1 Diagnostic Language Assessment (DLA)

Traditional language assessments typically focus on scoring overall learner ability, without analyzing specific strengths and weaknesses, and so may not be helpful to teachers or students (Poehner, Zhang & Lu, 2015). This issue is addressed with recent diagnostic language assessment (DLA) methods that implement a 3-stage process: 1) administration of the assessment; 2) provision of feedback to learners, and 3) further individualized instruction based on the assessment results. Therefore, the purpose of a

DLA is to assess learners' strengths and weaknesses, and provide tasks to help them improve. Practical DLA implementation faces three key challenges: 1) designing DLAs suitable for different learner skills; 2) identification of the most effective types of feedback for learners (Lee, 2015), and 3) efficient DLA administration (Poehner, 2008). One possible way to achieve efficiency is through computerization; however, typically this has meant learners receive only generic, rather than individualized, feedback (Harding, Alderson & Brunfaut, 2015).

2.2 English Email Writing and Pragmatic Competence

A faculty survey carried out at the higher education institution in which the current study is taking place has identified English email writing as a key task the undergraduate students should be able to perform (Kaneko et al., 2018). Upon graduating, this skill can also be beneficial for their future careers in a globalizing world. However, many students struggle with email writing in English. In particular, the pragmatic aspect of communication- in which social contextual variables such as the interlocutor's relative social status, social distance (how well the interlocutors know each other) and the potential imposition of the email communication upon the receiver (Brown & Levinson, 1987) can affect language choices related to politeness, formality and structure. This can be especially challenging for EFL learners (LoCastro, 2012). In addition, the pragmatic aspect of communicative competence is frequently undertaught in the EFL classroom; when it is addressed, it is often taught in an ineffective manner (McConarchy & Hata, 2013). While there are now various tools available that can help learners with the formal aspects of language such as grammatical difficulties (Grammarly or Google Translate for example), there is little help available for the pragmatic element of email communication. Further, while recently there have been efforts to develop DLA for reading and listening comprehension (Poehner et al., 2015; Yang & Qian, 2019), little DLA-related research has been conducted investigating writing ability. Specifically, applying DLA principles to communicative writing in varying social contexts, such as in email writing, has yet to be explored.

2.3 DLA and EFL email writing

We suggest focusing on two elements of this problem with pragmatic competence: 1) identifying precisely which pragmatic aspects of email writing learners find problematic; and 2) helping the learners improve their performance. Within the context of classes with large enrollments, the only feasible way to provide assessment and feedback is to implement a computerized DLA system; further, it should be able to provide individualized, specific feedback. However, little research has been done on developing a DLA of email writing skills that can help address problems students have with adapting their language choices to suit different social situations.

3. Developing a Basic DLA Prototype for EFL Learners' Email Writing

3.1 Context

As part of a larger study of DLA, Japanese EFL learners and email writing, the development of a basic computerized DLA prototype is one of the central tasks of the project. Figure 1 shows the overall development process for the DLA. The development of this prototype is the first part of the fourth step, Build DLA.

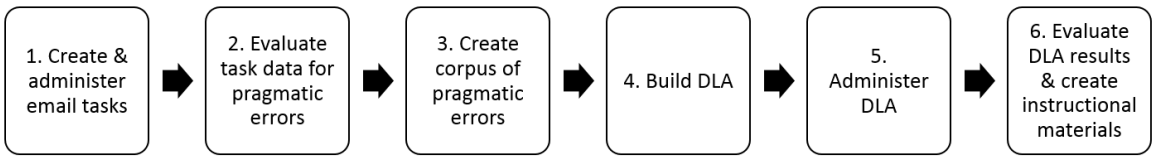

Figure 43: Core phases of the DLA development process

3.2 Rationale for Rapid Prototyping

Development of full-working versions of complex software packages is costly in terms of time, labor and finance. Prototypes or mock-ups are frequently used to gauge requirements, test design ideas and elicit feedback from stakeholders. Many organizations develop prototypes that can be trialed with stakeholders before starting work on the actual code. This enables developers to test the depth and breadth of their understanding of the requirements prior to developing fully-fledged code (Käpyaho & Kauppinen, 2015).

According to Tate (2010, n.p.), there is a "continuum of fidelity" that needs to be considered, starting from sketching concepts on paper to visual mock-ups and on to functional prototypes. The two most common types of rapid prototypes are low-fidelity and high-fidelity. Low-fidelity prototypes are very quick to create, but may lack either functionality or design, while high-fidelity prototypes appear professional and are fully functional for a number of use cases. High-fidelity prototypes aim to mimic the functions that users experience. For example, when creating a simple submission form, event handlers, such as mouseover and onclick need to be replicated. This is achieved by creating states and using interactions to switch between them, thus mimicking website behavior. The main downside to high-fidelity prototypes is the amount of time needed to create complex interactions.

Rapid prototyping was selected to create a high-fidelity prototype of the DLA system due to it being fast and economical (Chasanidou, Gasparini, & Lee, 2015). There are three key advantages to rapid prototyping: the early discovery of issues, its use as a communication aid, and its ability to test user interfaces and user experience.

- Early discovery of issues
  Low-fidelity prototypes can enable developers to gain insight into basic issues much earlier. High-fidelity prototypes force stakeholders to make decisions on more complex implementation issues earlier in the process.

- Communication aid
  Another advantage of rapid prototyping is that it is easy to show the software developer of the fully-functional version how the users are expected to interact with the interface. This can save time and reduce miscommunication between the developers and the clients. The developers can see how the clients expect users to interact using the prototype, and then aim to re-create the same feature and/or functions in the actual version. In short, high-fidelity prototypes provide developers with "living specifications" and enable showing rather than telling.

- Tool to gain insight into the user-interface (UI) and user experience (UX)
  This initial test of the graphical user interface (GUI) can help identify aspects of the user experience that can be enhanced. Multiple aspects of the user interface need to be considered, including layout, theme and colour. UX focuses on the whole experience of using the website. Simply put, UI focuses on what the user sees while UX focuses on what the user feels. The instrumental, experiential and emotional aspects of a GUI combine together to create the UX (Hassenzahl & Tractinsky, 2006). UI can be visualized through sketches on paper and stakeholder feedback gained on those sketches, but it is difficult for users to imagine how they might feel when using a website if they cannot actually use it. With a high-fidelity prototype, users can provide detailed feedback on UX.

3.3 Procedure for Rapid Prototyping

Use case models were created to envisage how actors will interact with the DLA. Table 1 shows some of the use cases considered in the development of the rapid prototype. A primary persona specification was created to represent a typical user.

Table 1
Sample use cases for the Rapid DLA Prototype

Use Case	Actor	Task
1	Student	A new user registers to use the DLA and logs off
2	Student	An existing user takes a DLA task
3	Student	An existing user receives a report after the fifth DLA task

Based on the use case models and discussions with stakeholders, a brief requirements analysis audit was conducted and a software requirements specification (SRS) was drawn up. The required functionalities were then listed. The prototype was developed based on the SRS and the list of required functionalities, taking into account the primary persona specification.

3.4 Prototyping software: Axure RP 9

Axure RP 9, a desktop prototyping tool with a built-in cloud-based shared facility, was selected based on its availability, suitability and ease of use. Axure is one of the most popular prototyping software programs (Carter & Hundhausen, 2010). According to Axure Software Solutions Inc. (2016, n.p.), "86% of the Fortune 100" companies create prototypes with their software and over a million prototypes are shared on their cloud-based platform. Axure RP has an intuitive GUI (see Figure 2).

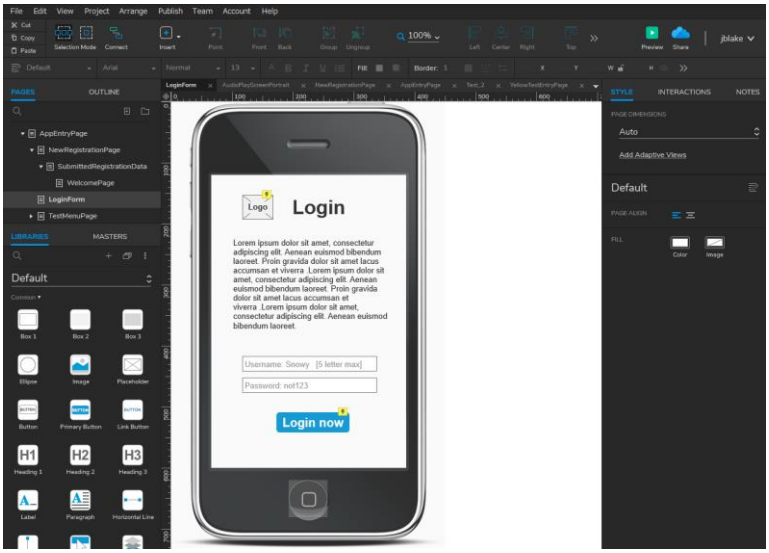

Figure 2. The user-interface for Axure RP showing a login screen

Creating a low-fidelity prototype of the DLA that can run in standard website browsers is straightforward when using this GUI. Figure 3 shows a screenshot of the login page in the high-fidelity prototype of the DLA created using Axure RP. The screenshot is taken from the use case in which the learner attempts the first DLA task. At this stage of this project, the nature of the tasks, expected pragmatic errors and content of the feedback are unknown. Placeholders of dummy text are used to provide a visual representation of the layout and quantity of text as seen through differently sized viewports on different devices.

4. Lessons learned

As with any new technology, novice users climb a learning curve. Five lessons were learned by the developer during the prototype design, development and initial *ad hoc* usability trials. The lessons described below aim to raise awareness of issues for other educators who may be considering prototyping their technology-enhanced language learning projects using Axure RP.

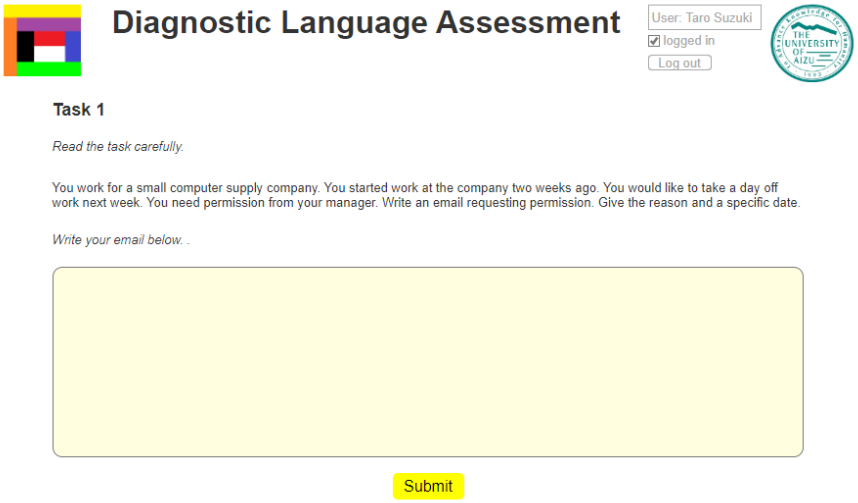

Figure 3. The prototype DLA GUI for task one created in Axure RP

4.1 One technology

When creating web-based tools, it is necessary to switch between various technologies, languages and libraries. For example, websites may use html, css, JavaScript and JQuery. PHP might be used to interact with a server and MySQL for a database. In Axure, however, the developer does not need to switch between different web technologies, reducing the need for mastering multiple technologies and speeding up the prototyping process.

4.2 Rapid prototyping is a misnomer

Low-fidelity rapid prototypes are aptly named, but high-fidelity rapid prototyping may take a considerable amount of time. The cost-benefit trade-off needs to be seriously considered if trying to replicate complex events. For example, the DLA system is required to automatically identify pragmatic errors that users make when writing in a submission form. To mimic this for multiple scenarios, it was necessary to create multiple regular expressions and make extensive use of states and interactions. A low-fidelity version, however, might only work for one use case which the user needs to input. Despite the apparent misnomer, the development of the prototype is rapid when compared to creating the full-code version.

4.3 Incomplete software requirements specifications

Some requirements that were needed by the developer were ambiguous while others were completely absent in the SRS. Should the full code have been outsourced using the identical SRS, this would undoubtedly have resulted in additional time and labor costs.

4.4 Usability issues with DLA prototype GUI

During the early stage of development, one Japanese speaker tested the interface, but struggled to understand what to do. This was because the instructions and interface were written in English only. Given that the DLA aims to test the use of English, this may be acceptable. However, as other users may face the same problem, the necessity for a bilingual interface needs further consideration.

4.5 User experience

The initial feedback from users was that the interface felt like a test. Although the DLA is an online assessment system in which users complete tasks, the purpose is to evaluate the their ability to write emails in English and then provide guidance based on their performance. Yet, the task looked more like a test rather than writing an email in a typical email client, such as Gmail, or Outlook Express. Features that Gmail affords, such as generated short responses, were absent. Users need to suspend disbelief and pretend that they are writing an email. Perhaps, it is viable to replicate an email client within the DLA.

5. Conclusion

The development of the DLA prototype was a steep learning curve for the developer, but was a very positive experience. Not only did the developer create a working prototype in a short timeframe, but there were many tangible outcomes that have a positive effect on the full-code version of the DLA. Most importantly the process forced numerous operational decisions to be made earlier than anticipated. Secondly, feedback on the UI and UX at such an early stage in this multi-year project means that resources can be dedicated earlier to improving these aspects. Thirdly, the prototype looks and behaves in a similar manner to the actual DLA and so can be used as an effective communication aid showing how the DLA works rather than describing how it should work. Therefore, it is suggested that the interactive nature of the prototype promotes understanding in a way that written descriptions or pictures cannot.

Acknowledgements

This research is supported by a JSPS KAKENHI (Grants-in-aid for Scientific Research), Grant Number 19K00826, "Developing a diagnostic language assessment of Japanese EFL learners' English language email writing".

References

Axure Software Solutions Inc. (2016, April 15). Axure RP 8 takes Prototyping & Diagramming to the Next Level. CISON PR Newswire.

Brown, P., & Levinson, S. C. (1987). *Politeness: Some Universals in Language Use*. Cambridge: Cambridge University Press.

Carter, A.S., & Hundhausen, C. D. (2010). How is User Interface Prototyping Really Done in Practice? A Survey of User Interface Designers. *Proceedings of 2010 IEEE Symposium on Visual Languages and Human-Centric Computing*, *38*, 207-211. https://doi:10.1109/vlhcc.2010.36

Chasanidou, D., Gasparini, A. A., & Lee, E. (2015, August). Design thinking methods and tools for innovation. In *International Conference of Design, User Experience, and Usability (pp. 12-23).* Springer, Cham.

Harding, L., Alderson, J.C., Brunfaut, T. (2015). Diagnostic assessment of reading and listening in a second or foreign language: Elaborating on diagnostic principles. *Language Testing 32*(3), 317-336.

Hassenzahl, M., & Tractinsky, N. (2006). User experience – a research agenda. *Behaviour & Information Technology*, *25*(2), 91-97. https://doi.org/10.1080/01449290500330331

Kaneko, E., Park, M., Wilson, I., Roy, D., Heo, Y., Yasuta, T., Blake, J., Nicholas, A. (2018). English curriculum innovation for computer science majors in the Japanese EFL context: From needs to tasks. *Proceedings of the 2018 IEEE International Professional Communication Conference*, 84-89.

Käpyaho, M., & Kauppinen, M. (2015, August). Agile requirements engineering with prototyping: A case study. In *Proceedings of the 2015 IEEE 23rd International requirements engineering conference (RE) (pp. 334-343).* IEEE.

Lee, Y-W. (2015). Diagnosing diagnostic language assessment. *Language Testing 32*(3), 299-316.

LoCastro, V. (2012). *Pragmatics for Language Educators: A Sociolinguistic Perspective*. New York: Routledge.

McConachy, T., & Hata, K. (2013). Addressing textbook representations of pragmatics and culture. *ELT Journal*, *67*(3), 294–301.

Poehner, M. (2008). Diagnostic Assessment: *A Vygotskian Approach to Understanding and Promoting L2 Development.* Berlin: Springer.

Poehner, M., Zhang, J., & Lu, X. (2015). Computerized dynamic assessment (C-DA): Diagnosing L2 development according to learner responsiveness to mediation. *Language Testing*, *32*(3), 337–357.

Tate, T. (2010, July 29). Concerning fidelity in design. UX Booth. https://www.uxbooth.com/articles/concerning-fidelity-and-design/

Yang, Y. & Qian, D. (2019). Promoting L2 English learners' reading proficiency through computerized dynamic assessment. *Computer Assisted Language Learning*, 1-25. https://doi.org/10.1080/09588221.2019.1585882