JavaScript for the Adult Novice

Catherine Sawyer
csawyer31@gatech.edu

***Abstract*—**The paper proposes a free asynchronous course in JavaScript for adult novices looking to change careers and enter the field of computer programming or other technologic areas. The course is based on current andragogical best practices, with scaffolded learning and just-in-time instruction that allows learners to work on a personally meaningful project. The authors compare the described course with existing resources for learning JavaScript and highlight the unique aspects of their proposal. The course aims to address the constraints faced by adult learners, such as time and financial limitations, and provide a more effective alternative to existing resources.

1 INTRODUCTION

There is a general sense among educators that computer programming is difficult for students to learn. This is particularly demonstrated by the fact that introductory coding classes have high attrition rates.

On the other hand, according to the US Bureau of Labor Statistics, Software Development jobs are expected to increase 25% from 2021-2031. The median salary for a software developer was over $109,000 in 2021, as compared to the US national median household income, which was about $70,000.

Because of these and other factors, there is a huge market for adult-centered 'upskilling' or job training in the area of computer programming. However, it is often difficult to find a reasonable resource for developing this skill. Firstly, the sheer volume of available learning opportunities can be overwhelming for a new learner. Professional working adults looking to switch careers and break into software development are faced with a huge number of potential learning opportunities; everything from YouTube and personal blogs to professional boot camps to programs like Georgia Tech's Online Master of Science in Computer Science (OMSCS), and everything in between.

The search for a "best fit" learning resource is further complicated by the fact that resources are not all of equal quality, varying by level of credibility and meaningfulness of credentials. Unfortunately, in addition to the number and the wide variations in quality of learning opportunities, many of these learning options are not well-fitted for the employed adult learner, as the training programs are often designed for traditional students who have the time and money to commit to a full-time program. This can be a barrier for many learners who have other commitments and who may not be able to commit to a pre-set schedule, and/or who can not afford the cost of a traditional program.

In light of these challenges, this project aims to provide a free, asynchronous course in JavaScript that is specifically designed for employed (or otherwise time-committed/limited) adult learners. The course is based on current andragogical best practices. It incorporates highly scaffolded learning and just-in-time instruction to help learners build their knowledge and skills in a graduated and effective manner. The course also focuses on helping learners develop a personally meaningful project or projects that they can use to demonstrate their skills to potential employers.

JavaScript has been identified in the literature as a good candidate for an introductory programming language. There are several factors that contribute to this identification, some of which include:

- JavaScript is relatively high-level and easy to understand
- It is a highly versatile language that has many applications
- It utilizes a syntax that is similar enough to other languages to make learning a second language somewhat intuitive

This project will therefore focus on helping adult learners become proficient in the use of JavaScript for computer programming. The aim is that by addressing the constraints faced by adult learners, a more effective alternative to existing resources will be provided that will help individuals break into technologic industries in the most efficient way possible.

2 RELATED WORK

There are a plethora of resources on how to learn programming, especially JavaScript. None of these resources, however, fulfill the needs of many adult

learners. Listed are a sampling of the types of resources found for learning JavaScript, and how they differ from the course described in this paper.

1. https://learnjavascript.online/
    a. LearnJavaScript.Online is a good resource for those who already have some rudimentary programming knowledge, but is not suitable for absolute novices. In addition, although it has many resources (such as a flashcard app) that will not be implemented in the course described in this paper, the projects used are "inspired by real-world projects", not actual self-determined projects, and most importantly, the learner can only get access to part of the content for free.
2. CodeAcademy Learn JavaScript
    a. This course covers only the very fundamentals of programming with JavaScript (up through how to craft a loop and use iterators), and although it advertises itself as free, only the lesson content is free, not any of the practice material in projects or quizzes. Without assessment, learning is hindered.
3. W3Schools JavaScript
    a. The free tutorial, while comprehensive, is not suited for beginners, and the full course (including certification) is locked behind a paywall.
4. Pluralsight Learn JavaScript
    a. Unclear what is taught in this course, as the learner has to pay for a full subscription to their course library to access the content.
5. JavaScript.Info
    a. This is an amazing resource, effectively a free living textbook online. There is a paid version for the EPUB (ebook) version, but the learner does not have to pay to access the online version. The major downside to this is that it doesn't have projects or practice problems; it is a resource, not a course.
6. LearnJavaScript.Today
    a. This course seems to accomplish many of the same tasks that the paper described in this paper is going to do, but LearnJavaScript is a synchronous course that is led by a single instructor/mentor, and is not a free course.

7. Boot camps, various (Flatiron Schools, Trilogy, General Assembly, ThriveDX, etc)
    a. These are very expensive courses. Unfortunately, almost all are synchronous, and they take extensive time to teach more skills than just JavaScript. While a good resource for those with the money and time to spend, this type of course does not work for many non-traditional (employed, time-committed) adult learners.

Browsing through this list, which presents samples of the majority of available types of learning resources, it can be seen that there is no ideal educational resource. As previously mentioned, there are two primary constraints faced by most nontraditional computer science/programming students, namely scheduled time commitments and cost, in addition to other constraints. As can be observed from the above list, which presents samples of the majority of available types of resources for learning JavaScript, these resources seem to all violate one or more of these constraints. Examples include:

- Cost
    - Many of the currently available resources involve significant cost. For example, boot camps charge an average of $20,000, and a learner may need to attend more than one "camp" to fully develop the skills that employers require. Many of the other available courses/resources are only fully available behind a paywall, and are therefore not accessible to all. While a good resource for those with the money to spend, these options do not work for everyone.
- Scheduled time commitment
    - Many of the courses/resources mentioned (CodeAcademy, W3Schools JS, Pluralsight, LearnJavaScript.Today, etc), are synchronous. Several (particularly the boot camps) add to the scheduled time commitment by spending significant time teaching more skills than just JavaScript. Again, this is useful for some adult learners, but not for many of them.
- Others
    - Many of the courses/educational resources sampled above introduce only the most basic of concepts rather than all of the skills necessary to progress in learning and gaining a job. (A few do offer more in-depth training, but only with paid, not free, access.) Others are poorly suited for learners trying to enter the

field from a non-technologic background, as they assume a degree of knowledge about computer programming and computer systems that are not applicable to novice learners.

In summary, there is no readily identifiable learning resource that is fully free (no charge), that is asynchronous and adaptable to any learner's schedule, that assumes no prior knowledge of computer programming, but is able to start a novice learner on the most basic, introductory level of computer programming, and guide them in a systematic manner (based on best adult-learner andragogic methods) through all necessary levels of learning that will bring that learner to the point of being skilled enough to work in some aspect as a computer programmer. The course presented in this paper fills that void and provides the described resource.

3 SOLUTION

This project aims to help adult novices attempting to break into programming and technologic fields by providing a free, project-based, highly scaffolded, asynchronous JavaScript course. The course is composed of four modules. Each module is broken down into several one-hour lessons, each with specific learning objectives, practical examples, and acceptance criteria for the completion of a practical project related to the presented material. The course syllabus is as follows.

3.1 Syllabus

Module 1: Introduction, tools setup, basic web design, and an overview of JavaScript - Get started, get used to the course structure, and set up one's computer to start programming. Create a website that utilizes eventListeners buttons, and basic forms to incorporate interactivity.

Example project: Create an informational website with a trivia game.

Module 2: Basic Programming - Dig into some programming basics, including variables and data types, booleans and conditionals, operators and advanced loops.

Example project: Enhance one's website by moving the trivia game to its own tab, and adding another tab with a todo list functionality.

Module 3: Intermediate Programming - Learn about functions and classes, Object Oriented Programming (OOP) and Functional Programming paradigms.

Example project: Choose OOP or Functional programming to enhance the functionality of the trivia game or todo list.

Module 4: Advanced Programming Concepts - Dive into some further programming concepts, including recursion, error handling, and asynchrony in JavaScript.

Example Project: Add another tab to one's webpage with a web-based game such as breakout, a maze solver, or a click-based RPG game.

At the end of the course, the learner will feel confident in their ability to code in JavaScript, and in their ability to continue to learn more as needed by navigating online resources. They will have gained practical knowledge of how to read and understand code, generate code from scratch, to read and use documentation and tutorials, and to break problems down into smaller pieces in order to solve them.

In addition to building the learner's knowledge and skills in JavaScript, this course will help the learner develop a personally meaningful project that they can use to demonstrate their abilities to potential employers. Throughout the course, the learner will be given guidance and support to help them identify a project idea that is relevant and interesting to them. The lessons will present material to help them build the skills and knowledge they need to implement their project. Each lesson in the course builds on the previous one, so that the learner is provided with the tools and resources they need to make progress on their project. By the end of the course, they will have a fully-functional project that they can use to showcase their abilities and that will help them break into the technologic industries.

4 METHODOLOGY

Scaffolding the course content appropriately is one of the most important characteristics of this course, differentiating it from other courses. Rather than a strictly linear learning path, this course focuses on making sure that learners are able to apply skills that they learn in a consistent way, so that skills build on each other towards a bigger project.

In addition to providing a more effective learning experience, this not-strictly-linear learning path also has the advantage of better preparing learners for real-life projects. In technologic industries, projects are rarely strictly linear and often require learners to apply multiple skills and knowledge areas in order to complete them.

By providing learners with the opportunity to build their skills in this way, the course helps them develop the ability to adapt to changing requirements and solve complex problems. This is a valuable skill that is essential for success in any technology industry, and one that is not always emphasized in traditional learning paths. By incorporating a not-strictly-linear learning path, the course aims to better prepare learners for the demands of technologic industries and help them succeed in their career aspirations.

To illustrate this andragogical technique, strictly linear learning path is contrasted with the content of the course described in this paper:

For example: to teach someone how to use JavaScript, the following topics would need to be covered:

1. Introduction to programming concepts and the basics of JavaScript syntax
2. Working with data types and variables
3. Using loops and conditional statements to control the flow of a program
4. Creating and manipulating arrays and objects
5. Working with functions and object-oriented programming principles
6. Using JavaScript to interact with web pages and create interactive websites
7. Debugging and error handling
8. Introduction to more advanced concepts such as asynchronous programming and working with APIs.

In a linear teaching path, learners would be introduced to data types and variables, then asked to practice using these concepts. Next, they would be introduced to conditional statements, and possibly taught how to create a rock-paper-scissors game or similar simple application.

With each step, learners would be given a small building block towards a greater understanding, but wouldn't necessarily be able to utilize those steps effectively to create and implement a meaningful or bigger project until a significant portion

of content had been covered. This tends to result in learners disengaging from the content and not finishing what they start, especially in an asynchronous environment.

In contrast, the described course's not-strictly-linear learning path introduces learners to JavaScript through event listeners. They are given example code to see how a website can be made interactive in a simple way, then instructed to edit the code in progressively more complicated ways. Next, they are introduced to variables, then arrays and for loops. At each step, they are given Acceptance Criteria to help them implement their skills into a real project, with an example project provided that will help the learner practice and cement their skills. The project that they work on also serves as a portfolio piece for them to demonstrate their skills to prospective future employers.

In addition to being an effective way to learn JavaScript, this approach also has the advantage of being more engaging and motivating for learners, because it gives learners the opportunity to work on a project that is personally meaningful to them. This is particularly important for adult learners, who may have other commitments and may not be able to devote as much time and energy to their learning as traditional students. By providing learners with a supportive and engaging learning environment, the course aims to help them succeed in their career aspirations and upskilling journey.

Another key aspect of the course that differentiates it from many other courses/educational offerings is the inclusion of "web quest" assessment questions and instruction. One of the key skills that software developers at large need to master is the ability to search for, find (largely on the internet), read and understand documentation that answers their questions.

To ensure that learners develop and master this skill, they are given at least one 'web quest' type assessment question in each lesson. The first such question explicitly instructs learners to search for specific terms, select a particular website from the results, and find a very specific image on that page. The next web quest instructs learners to find and use a color picker tool to choose and then use a hex color code in their website, and the next instructs them to search for a key term on a specific website.

The assessment questions get progressively less specific in telling learners how to do things, while maintaining the requirements on finding and using answers. In this way, learners develop skills in searching for and understanding answers to questions in addition to the code/programming skills that they are acquiring.

One final, key defining feature of the course is its use of projects in cementing learning. By utilizing the programming skills that are introduced in each lesson to build something, learners are able to make use of the skills introduced, practice planning and building larger applications, and experience debugging and problem solving in a way that relates to bigger projects. In addition, the inclusion of projects helps to keep learners engaged, as they can be working towards finishing something that is personally meaningful, and ultimately will help to demonstrate the skills that they have learned to future employers.

5 RESULTS

Currently, the described course content is being authored and is partially developed. The source code for the website where the lesson content is hosted, as well as the lesson markdown text files, can be found on GitHub here: https://github.com/crsawyer314/crsawyer314.github.io, and the published website can be found here: catherinesawyer.netlify.app.

The lesson outlines can be found here:

https://docs.google.com/document/d/1Jz7aembJ9iljtugzim9AlLxfQnO-FjFxLBPyo-ipMtM/edit?usp=sharing.

Feedback, utilizing a convenience sample of novice users who have reviewed course content, structure, and presentation, has been universally positive. These results suggest that the described course is an effective way to teach adult novice learners the basics of JavaScript. Reviewers particularly appreciated the course's asynchronous format, which allows them to complete the course at their own pace. These reviewers commented that this allows them to effectively balance their learning with their other time-commitments.

The scaffolded learning approach, which provides learners with just-in-time instruction and support, was also well-received by reviewers, as it allows learners to gradually build their knowledge and skills, starting with the basics and gradually working up to more complex concepts.

As previously mentioned, one of the unique aspects of the described course is its focus on helping learners develop a personally meaningful project. This was an important aspect of the course for reviewers, as it allows them to apply their newly acquired knowledge to a real-world problem, and to create something that could be shown to potential employers as part of their job-hunting efforts.

The early feedback and reviews of the described course also provided insight into the challenges that reviewers anticipate learners will face in the course. The most commonly anticipated challenges were related to the self-paced nature of the course, as some reviewers felt it would be difficult to stay motivated and focused without the support of a traditional classroom setting. To address this issue, eventually more support and accountability mechanisms will be incorporated into the course, such as regular automated check-ins and progress tracking, to help learners stay on track and engaged. This sort of progress tracking could be implemented by implementing an account on the website with settings for regular notifications at user-specified intervals. This would allow the course to remain asynchronous while also helping learners to stay engaged.

Overall, the results of the early feedback indicate that the described course is a promising way to teach adult novices the basics of JavaScript and to provide them with the skills and knowledge they need to break into technologic industries. Further research will be needed to confirm the effectiveness of the course and to identify potential areas for improvement.

6 LIMITATIONS

Once completed, the largest limitations that the project faces generally center around the scope of what will or can be offered. Due to the nature of the content, no credential will be available, and there is no guarantee that learners will finish the content once they've started it. By offering the course for free to anyone, the knowledge will be available for anyone to learn, but it may be a challenge to get and implement feedback on the course content without a way of tracking learners as they use the materials.

It is also difficult to know how far-reaching this course content will be. Curricula and education in general is most effective when it is actually used, so investing a significant amount of time to create a course that ultimately collects dust in the back corner of the internet would be less than desirable. And without charging

for the content there won't be funding to cover the cost of promoting the course on search engines or elsewhere.

7 CONCLUSION

The described JavaScript course aims to address the challenges faced by adult learners looking to break into technology industries, such as time and financial constraints, as well as the need to demonstrate their skills to potential employers. The course is designed to be asynchronous, allowing learners to complete it at their own pace, and is offered for free. The course is highly scaffolded, providing learners with just-in-time instruction and support for personal projects as they progress through the material. Overall, the described course provides a valuable resource for adult learners looking to learn programming and to break into technologic industries.

8 FUTURE WORK

In order to evaluate the effectiveness of the described course, a pilot study will be conducted with a small group of volunteer adult learners who have no prior experience with programming. The study will be conducted over a period of six weeks, during which participants will be provided with access to the course materials and be instructed to complete the course at their own pace.

At the end of the study, participants will be asked to complete a survey to evaluate the course and provide feedback on their learning experience. So far, feedback on the course content has been overwhelmingly positive. It is hoped that the results of this pilot study will be similarly positive.

In addition to the survey, a series of interviews will be conducted with the participants to gather detailed individualized feedback on the course. The interviews will provide insight into the challenges that participants face while participating in the course, and the strategies used to overcome said challenges.

Overall, the results of the pilot study will give feedback on the effectiveness of the described course in teaching adult novices the basics of JavaScript. The pilot study will help to further refine the content to be most effective in helping adult novices to break into the "tech" industry.

9 REFERENCES

1. Arawjo, I., Wang, C. Y., Myers, A. C., Andersen, E., & Guimbretière, F. (2017, May). Teaching programming with gamified semantics. In Proceedings of the 2017 CHI conference on human factors in computing systems (pp. 4911-4923).
2. Bohlin, R. M., & Milheim, W. D. (1994). Applications of an Adult Motivational Instructional Design Model.
3. Bosse, Y., & Gerosa, M. A. (2017). Why is programming so difficult to learn? Patterns of Difficulties Related to Programming Learning Mid-Stage. ACM SIGSOFT Software Engineering Notes, 41(6), 1-6.
4. Crockford, D. (2001). JavaScript: The world's most misunderstood programming language. Douglas Crockford's Javascript.
5. Etim, G. L., Chinwuba, C. C., & Iyamu, I. (2020). Coding: Simplified Approach and Techniques for Beginner–Programmers. International Journal of Innovative Science and Research Technology, 5.
6. Giguette, R. (2003). Pre-games: games designed to introduce cs1 and cs2 programming assignments. ACM SIGCSE Bulletin, 35(1), 288-292.
7. Guzdial, M., & Soloway, E. (2002). Teaching the Nintendo generation to program. Communications of the ACM, 45(4), 17-21. https://dl.acm.org/doi/fullHtml/10.1145/505248.505261
8. Horváth, G., & Menyhárt, L. (2014). Teaching introductory programming with JavaScript in higher education. In Proceedings of the 9th International Conference on Applied Informatics (Vol. 1, pp. 339-350).
9. Kim, J., & Lerch, F. J. (1997). Why is programming (sometimes) so difficult? Programming as scientific discovery in multiple problem spaces. Information Systems Research, 8(1), 25-50.
10. Mahmoud, Q. H., Dobosiewicz, W., & Swayne, D. (2004). Redesigning introductory computer programming with HTML, JavaScript, and Java. ACM SIGCSE Bulletin, 36(1), 120-124.
11. MDN Contributors. (2019). JavaScript. MDN Web Docs. https://developer.mozilla.org/en-US/docs/Web/javascript
12. Minnick, C., & Holland, E. (2015). Coding with JavaScript for dummies. John Wiley & Sons.
13. Nandigam, D., & Bathula, H. (2013). Competing dichotomies in teaching computer programming to beginner-students. American Journal of Educational Research, 1(8), 307-312.

14. Nishide, N. S. I. P. R., & Takada, H. (2017). An interactive environment to teach programming based on chain learning. 情報処理学会第 79 回全国大会, 2, 02.
15. Ravitz, J., Mergendoller, J., Markham, T., Thorsen, C., Rice, K., Snelson, C., & Reberry, S. (2004). Online Professional Development for Project Based Learning: Pathways to Systematic Improvement. Beryl Buck Institute for Education. https://www.academia.edu/73037479/Online_Professional_Development_for_Project_Based_Learning_Pathways_to_Systematic_Improvement
16. Shrestha, N., Botta, C., Barik, T., & Parnin, C. (2020, October). Here we go again: Why is it difficult for developers to learn another programming language?. In 2020 IEEE/ACM 42nd International Conference on Software Engineering (ICSE) (pp. 691-701). IEEE.
17. "Software Developers, Quality Assurance Analysts, and Testers : Occupational Outlook Handbook." U.S. Bureau of Labor Statistics, U.S. Bureau of Labor Statistics, 9 Sept. 2022, https://www.bls.gov/ooh/Computer-and-Information-Technology/Software-developers.htm.
18. Szlávi, P., & Bakonyi, V. H. (2014) Evaluation of programming languages from educational perspective.
19. Viriyadamrongkij, N., & Senivongse, T. (2017, July). Measuring difficulty levels of JavaScript questions in Question-Answer Community based on concept hierarchy. In 2017 14th International Joint Conference on Computer Science and Software Engineering (JCSSE) (pp. 1-6). IEEE.