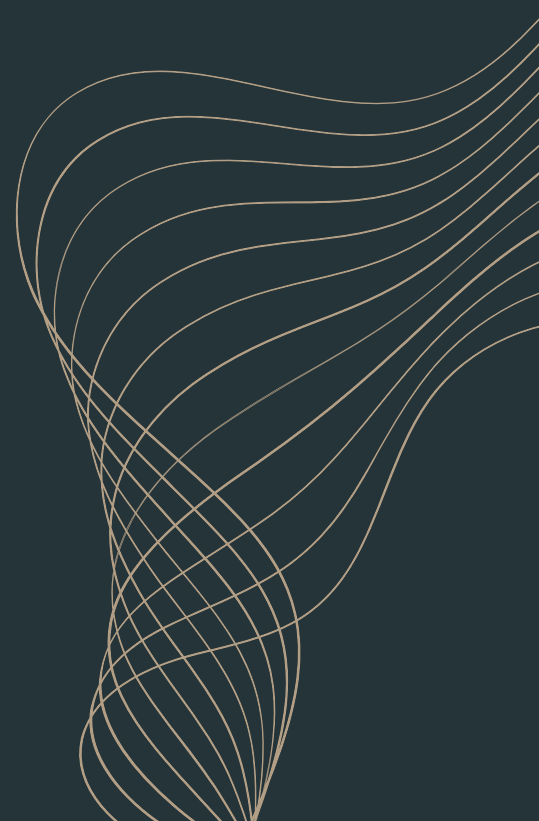




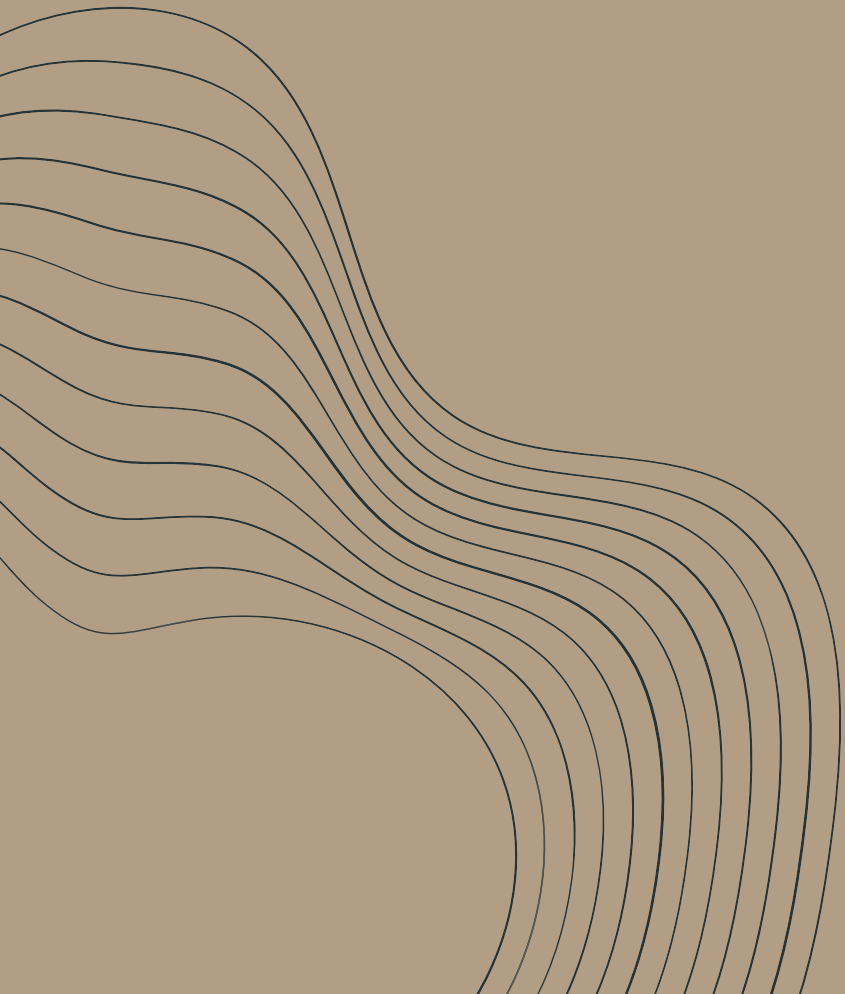
# RIVERS RISE E-COMMERCE WEBSITE

Chamberlain L. Gibson  
Cybersecurity  
Project Advisor: Julie Henderson



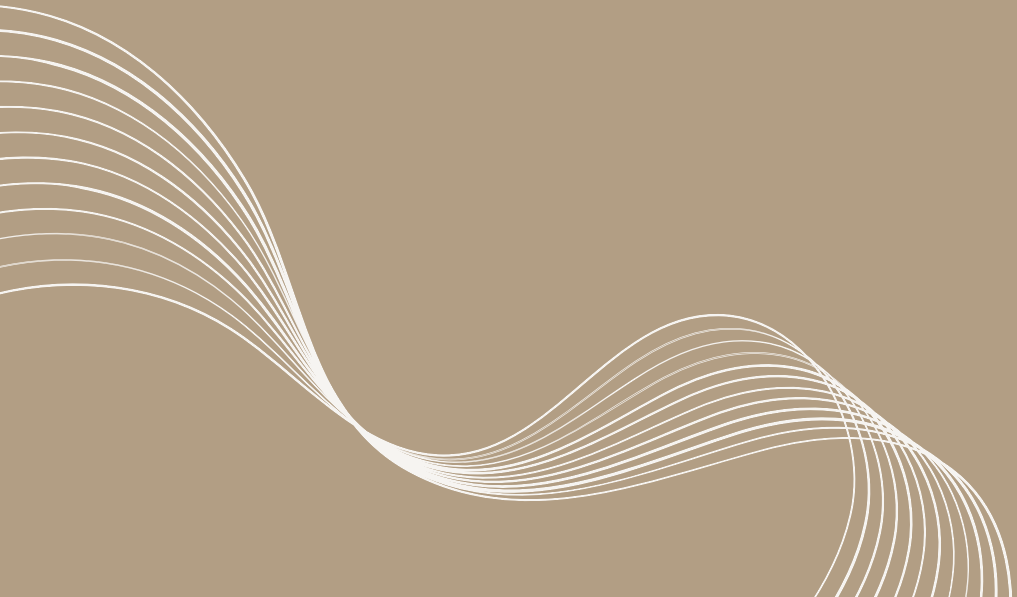


# CONTENTS

- 
- |    |   |
|----|---|
| 01 | Statement of Purpose                    |
| 02 | Research & Background                   |
| 03 | Project Implementation                  |
| 04 | Project Languages, Hardware, & Software |
| 05 | Tests Results                           |
| 06 | Challenges Overcome                     |
| 07 | Future Enhancements                     |

# STATEMENT OF PURPOSE

- To give my client access to a website and cloud-based database that upholds a sophisticated level of security and usability to support the expansion of their business.



# RESEARCH & BACKGROUND

## WEBSITE STRUCTURE



- Small business platforms for development purposes.
- Building hypothetical websites for future improvement.
- Integrating familiar web-based programming languages.

- Databases that connect to front-end development.
- Client administrative access for product addition.
- Database table populating to website.
- Customer information storage from sign-in page.

## DATABASE STRUCTURE

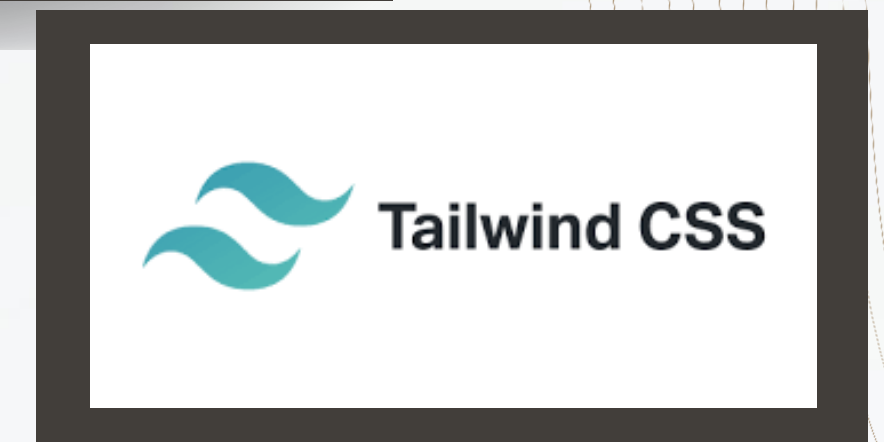


# PROJECT IMPLEMENTATION

- Live Demo

# PROJECT LANGUAGES, SOFTWARE, HARDWARE

- Implementation Languages: Ruby on Rails, HTML, JavaScript
- Libraries, Packages, Development Kits: Devise, Turbo, Hotwire, Tailwind CSS, CarrierWave, MiniMagick, ImageProcessing, Paperclip, Capybara, Stimulus, Stripe, PostgreSQL, Pagy, Brakeman, Bundler-audit
- Software/Equipment Needed: VMWare Virtual Machine





# TEST RESULTS

- Usability Tests Overview
  - Evaluate product's ease of use and user-friendliness.
  - Involves observing real users' interactions.
  - Identifies usability issues or improvement areas.
  - Includes task definition, user observation, feedback collection, and analysis.
  - Aids in understanding user perception and interaction.
  - Leads to enhancements improving user satisfaction and experience.
- Brakeman Configuration:
  - Configure Brakeman settings, including any custom rules or exclusions tailored to the applications requirements.
  - Ensure that Brakeman is integrated into the development workflow, such as being included in continuous integration pipelines.
  - Run Brakeman against the Rails application to scan for security vulnerabilities.
  - Review the generated report to identify any potential issues detected by Brakeman.
  - Prioritize and categorize the identified vulnerabilities based on their severity and impact on the application.

# CHALLENGES OVERCOME

## TIME MANAGEMENT

- Overcame time management issues due to self-reliance.
- Learned about work prioritization and self-imposed deadlines.
- Divided project into smaller, achievable tasks.
- Constructed a timeline for timely completion.

## DEVELOPMENT

- Project surpassed User Interface Programming course's scope.
- Ruby on Rails lesson was complex due to software applications like Devise.
- Devise encryption was beneficial for security.
- Steep learning curve but rewarding experience.
- Enhanced programming skills through increased knowledge and proficiency.

## SECURITY

- Project surpassed User Interface Programming course's scope.
- Ruby on Rails lesson was complex due to software applications like Devise.
- Devise encryption was beneficial for security.
- Steep learning curve but rewarding experience.
- Enhanced programming skills through increased knowledge and proficiency.



# FUTURE ENHANCEMENTS

- Quality User Interface
  - Improved brainstorming sessions within a software development team for better interface choices.
  - Effective time management in a group setting for better collaboration.
  - Innovative ideas and creative solutions arising from team collaboration.
  - Utilization of diverse skills and perspectives for a visually appealing, functional, and user-friendly user interface.
- Website Security Measures Improvement
  - Increased time management and robust website and database.
  - Implementation of encryption protocols and regular security audits.
  - Strengthening user authentication processes to prevent unauthorized access.
  - Enhancement of user experience and trust building.
  - Regular monitoring and updates to stay ahead of emerging threats.
  - Prioritization of security and user privacy to maintain website's reputation as a reliable and secure platform.

# REFERENCES

- Blaz JW;Pearce PF. (2024). Using ruby on rails to develop a web interface: a research-based exemplar with a computerized physical activity reporter. *Studies in Health Technology and Informatics*, 146. <https://pubmed.ncbi.nlm.nih.gov/19592849/>
- chart.js. (2023, December 4). Npm. <https://www.npmjs.com/package/chart.js?activeTab=readme>
- DevGraph. (2021, April). Ruby On Rails Web Development. DEV Community; DEV Community. <https://dev.to/devgraph/ruby-on-rails-web-development-1b1d>
- Dragonfly. (2024). Github.io. <https://markevans.github.io/dragonfly/>
- File: README — Documentation for teamcapybara/capybara (master). (2024). Rubydoc.info. <https://rubydoc.info/github/teamcapybara/capybara#using-capybara-with-rspec>
- Fullstack E-Commerce: Ruby on Rails 7, Hotwire, Tailwind, Stripe, PostgreSQL. (n.d.). Ww.youtube.com. Retrieved February 19, 2024, from <https://www.youtube.com/watch?v=hURUMwdCWul&t=1109s>
- Gleb B, & Vlad V. (2019, July 13). Ruby on Rails Web Application Vulnerabilities: How to Make Your App Secure. Rubygarage.org; RubyGarage. <https://rubygarage.org/blog/ruby-on-rails-web-application-vulnerabilities-how-to-make-your-app-secure>
- How. (2024, January 9). How to Use Brakeman to Find Rails Security Vulnerabilities - FastRuby.io | Rails Upgrade Service. How to Use Brakeman to Find Rails Security Vulnerabilities . <https://www.fastruby.io/blog/how-to-use-brakeman-to-find-rails-security-vulnerabilities.html#:~:text=To%20run%20Brakeman%2C%20navigate%20to,security%20issues%20and%20their%20severity.>
- How To Set Up User Authentication with Devise in a Rails 7 Application | DigitalOcean. (n.d.). Ww.digitalocean.com. <https://www.digitalocean.com/community/tutorials/how-to-set-up-user-authentication-with-devise-in-a-rails-7-application>
- Labs, X. (2014). Rails Best Practices - Rails Best Practices. Rails-Bestpractices.com. <https://rails-bestpractices.com/>
- Learn Hotwire and Turbo with a free Rails 7 tutorial. (n.d.). Learn Hotwire and Turbo with a Free Rails 7 Tutorial. <https://www.hotrails.dev/turbo-rails>
- OWASP Secure Headers Project | OWASP Foundation. (2015). Owasp.org. <https://owasp.org/www-project-secure-headers/>
- Paul, A., Philips, P., Dániel Szabó, & Wallace, C. (2019). *The Ruby Workshop A Practical, No-Nonsense Introduction to Ruby Development*. Birmingham Packt Publishing Ann Arbor, Michigan Proquest.
- PostgreSQL Tutorial - Learn PostgreSQL from Scratch. (n.d.). <https://www.postgresqltutorial.com/>
- Przemysław Łata. (2022, September 13). Reasons for choosing Ruby on Rails for e-commerce projects. Railwaymen.org; Railwaymen Sp. z o.o. <https://blog.railwaymen.org/ruby-on-rails-for-e-commerce#point-1>
- Roller, J. (2022, August 30). A Developer's Insight into The Pros and Cons of Ruby on Rails. IEEE Computer Society. <https://www.computer.org/publications/tech-news/build-your-career/developers-insight-into-the-ruby-on-rails>
- Ruby on Rails API. (2024). Rubyonrails.org. <https://api.rubyonrails.org/>
- Ruby on Rails Guides. (2023). Ruby on Rails Guides. <https://guides.rubyonrails.org/>
- Stimulus Handbook. (n.d.). Stimulus.hotwired.dev. Retrieved February 19, 2024, from <https://stimulus.hotwired.dev/handbook/origin>
- Syed Fazle Rahman. (2015). *Bootstrap for Rails*. Packt Publishing Ltd.
- Tailwind CSS Login / Sign In Form - Free Examples & Tutorial. (n.d.). TW Elements. Retrieved February 19, 2024, from <https://tw-elements.com/docs/standard/forms/login-form/>
- tailwindcss. (2023). Tailwind CSS - Rapidly build modern websites without ever leaving your HTML. Tailwindcss.com. <https://tailwindcss.com/>
- Viswanathan, V. (2008). Rapid Web Application Development: A Ruby on Rails Tutorial. *IEEE Software*, 25(6), 98–106. <https://doi.org/10.1109/ms.2008.156>
- (2023). Stripe.com. <https://docs.stripe.com/>



QUESTIONS?