February 22, 2024

C964: Computer Science Capstone Template

Task 2 parts A, B, C and D

[Part A: Letter of Transmittal 2](#_Toc159513543)

[Part B: Project Proposal Plan 4](#_Toc159513544)

[Project Summary 4](#_Toc159513545)

[Data Summary 4](#_Toc159513546)

[Implementation 6](#_Toc159513547)

[Timeline 6](#_Toc159513548)

[Evaluation Plan 6](#_Toc159513549)

[Resources and Costs 6](#_Toc159513550)

[Part C: Application 7](#_Toc159513551)

[Part D: Post-implementation Report 8](#_Toc159513552)

[Solution Summary 8](#_Toc159513553)

[Data Summary 8](#_Toc159513554)

[Machine Learning 8](#_Toc159513555)

[Validation 8](#_Toc159513556)

[Visualizations 8](#_Toc159513557)

[User Guide 8](#_Toc159513558)

[Reference Page 10](#_Toc159513559)

# Part A: Letter of Transmittal

February 22, 2024

Ronald MacDonald

Translatinator, LLC

123 Sunshine Road Los Angeles CA, 90001

Dear Ronald MacDonald

I hope this message finds you well. I am writing to propose an exciting enhancement to our language-learning application, Translatinator, which I believe will significantly distinguish our product from competitors such as RosettaStone and DuoLingo. Despite offering a high-quality product at an excellent price point, Translatinator currently lacks unique features that set it apart in the crowded language-learning space.

**Proposal Overview:**

I propose the integration of a cutting-edge feature that leverages machine learning technology. This feature will enable users to take pictures of common objects, specifically cats and dogs in the initial phase, and receive the name of the object in the language they are learning. This approach not only adds a novel functionality to our app but also aligns with our mission to provide intuitive and innovative language-learning solutions.

**Benefits to Translatinator:**

***Enhanced Functionality***: This feature will enrich the user experience by offering a practical, real-world application of language learning, thus increasing user engagement and satisfaction.

***Technological Relevance***: By incorporating machine learning, we underscore our commitment to innovation, keeping Translatinator at the forefront of technology trends in educational applications.

***Intuitive Learning***: This enhancement aligns with our goal to create a natural and immersive learning environment, making the acquisition of new vocabulary more intuitive and engaging.

**Project Summary:**

***Costs***: The project entails 200 hours of developer time at $40/hour and 100 hours of project management at $30/hour, with additional costs for server usage at $500/month to train and store the AI model. The software framework, TensorFlow, is available at no cost.

***Timeline***: We aim to complete this project within 6 weeks, ensuring a swift enhancement to our app's capabilities.

***Data***: The development will utilize 3,500 labeled images of cats and dogs, providing a robust dataset for initial training.

**My Expertise:**

As a computer science student with extensive experience in machine learning applications, including proficiency in Python, AIML, and application design, I am well-equipped to lead this project to success. My background ensures a deep understanding of the technological and practical aspects necessary for this enhancement.

I am confident that this feature will not only improve our competitive edge but also demonstrate our ongoing commitment to innovation and customer value. I look forward to discussing this proposal further and am available for any questions or additional information you may require.

Thank you for considering this proposal. I am excited about the potential impact this project could have on Translatinator’s market position and our users' learning experiences.

Sincerely,

Andrew Steurer

Andrew Steurer, Intern

# Part B: Project Proposal Plan

The project proposal should target your client’s middle management. This audience may be IT professionals but have limited computer science expertise. Use appropriate industry jargon and sufficient technical details to describe the proposed project and its application. Remember, you’re establishing the technical context for your project and how it will be implemented for the client. **Write everything in the future tense.**

## Project Summary

* **Summary of the Problem**
  + Translatinator, a competitive language-learning application in the market alongside giants like RosettaStone and DuoLingo, presents a high-quality product at an exceptional price. Despite these advantages, Translatinator struggles to stand out due to its lack of distinctive features. This challenge hampers its ability to attract new users and retain existing ones, necessitating a solution that enhances its appeal and functionality.
* **Company Needs**
  + Translatinator requires a novel approach to elevate their market position and differentiate it from the competition. The core need is to integrate a feature that not only adds value to the user experience but also leverages current technological trends to enhance learning outcomes.
* **Deliverables**
  + ***Machine Learning Model***: A robust model will be trained to recognize images of cats and dogs with high accuracy.
  + ***Application***: A command line interface that will allow the user to interact with the machine learning model.
  + ***User Guide***: A set of instructions for how to set up the environment and how to run the application.
* **Summary of Benefits** 
  + ***Added Functionality***: By enabling users to learn words through object recognition, Translatinator will add a unique, interactive element to language learning that is currently absent in competitor apps.
  + ***Technological Relevance***: Incorporating machine learning for object recognition signals to current and potential users that Translatinator is at the forefront of applying emerging technologies to enhance learning experiences.
  + ***Intuitive Learning Experience***: This feature will offer a more natural, engaging way for users to learn new vocabulary, mimicking the way languages are learned naturally through interaction with the environment.

## Data Summary

* The categorized image data will be sourced from two data repositories hosted on kaggle.com:
  + <https://www.kaggle.com/datasets/alifrahman/dataset-for-wbc-classification>
  + <https://www.kaggle.com/datasets/samuelcortinhas/cats-and-dogs-image-classification>
* Development Life Cycle Phases:
  + Design:
    - Data Requirement Analysis: Identify the types of data needed for the machine learning model, specifically focusing on images of cats and dogs. This includes understanding the diversity in breeds, sizes, and environments to ensure the model is well-trained.
    - Data Privacy and Security Assessment: Evaluate the implications of handling user-uploaded images, ensuring compliance with privacy regulations. This involves designing data anonymization processes and secure data storage solutions.
    - Data Collection Plan: Outline methods for collecting training data, ensuring it's ethically sourced and respects copyright laws.
  + Development:
    - Data Preparation and Preprocessing: Clean, annotate, and prepare the collected images for training. This process includes categorizing images, removing identifiable information, and possibly augmenting the dataset to improve model robustness.
    - Model Training and Testing: Utilize the processed data to train the machine learning model. This involves iterative testing and validation against a test dataset to ensure accuracy and minimize bias.
    - Application Integration: Build a command line interface that interacts with the model and allows testers to provider their own images to test.
  + Maintenance:
    - Continuous Data Monitoring: Implement monitoring tools to track the performance of the object recognition feature and identify any issues with data processing or model accuracy.
    - Data Storage Management: Regularly review data storage practices to ensure they remain secure and compliant with evolving data protection laws. This includes periodic audits and updates to data encryption and anonymization techniques.
    - Model Updates and Re-training: As new data becomes available or user feedback highlights areas for improvement, re-train the model with new images or refined algorithms. This may include expanding the model's capabilities to recognize additional objects beyond cats and dogs.
* Justify why the data meets the needs of the project.
  + The core functionality of the new feature is to recognize images of common objects (initially, cats and dogs) and translate their names into the language a user is learning. Thus, a dataset comprising diverse images of cats and dogs is directly relevant as it serves as the foundation for training the machine learning model to accurately identify these objects. Focusing on cats and dogs makes the project more manageable and achievable within its scope. It ensures that the data collection, processing, and model training phases are practical and feasible, providing a clear pathway to successful implementation and deployment. This focus allows for a concentrated effort on achieving high accuracy and user satisfaction with the feature before expanding to more complex objects or categories.
* Address any ethical or legal concerns regarding the data.
  + There are no ethical concerns with the current data. The data is sourced from a free, public data repository that contains images of dogs and cats only.

## Implementation

* Describe an industry-standard methodology to be used.
  + For the Translatinator enhancement project, implementing the feature that allows users to take pictures of common objects (initially cats and dogs) and receive the translation in the language they are learning, an industry-standard methodology that would be particularly effective is Agile development with Scrum. This methodology is well-suited for projects requiring flexibility, rapid iteration, and close collaboration between developers, project managers, and stakeholders. Agile development emphasizes iterative progress, flexibility, and the delivery of functional product components in short cycles called sprints. Scrum, a subset of Agile, provides a structured framework for managing this process, focusing on quick deliveries, and incorporating feedback to refine and adjust the project direction as needed.
* An outline of the project’s implementation plan.
  + Phase 1: Pre-Development
    - Define the scope and specific objectives for the machine learning solution.
    - Identify the languages and the number of objects (initially cats and dogs) to be recognized.
    - Define the scope and specific objectives for the machine learning solution.
    - Identify the languages and the number of objects (initially cats and dogs) to be recognized.
  + Phase 2: Development
    - Set up the development environment and tools.
    - Develop initial prototypes of the machine learning model using a subset of the data.
    - Evaluate prototype models to select the most promising approach.
    - Expand model training using the full dataset.
    - Implement model validation and tuning to improve accuracy and reduce overfitting.
    - Build a CLI application that interacts with the model.
  + Phase 3: Testing
    - Conduct comprehensive testing including unit tests, integration tests, and user acceptance testing.
    - Collect feedback from testers to identify issues and areas for improvement.
  + Phase 4: Deployment and Monitoring
    - Finalize documentation.
    - Deploy the enhanced feature to the production environment.
    - Monitor the feature's performance and user feedback closely.
    - Address any immediate issues or bugs reported by users.
  + Phase 5: Improvements
    - Continue to iterate on the machine learning model and feature based on ongoing user feedback and technological advancements.
    - Plan for the expansion of the object recognition capability to include more objects beyond cats and dogs, based on user demand and success metrics.

## Timeline

* Provide a projected timeline, including projected start dates and end dates for each milestone (a table is not required but encouraged).

|  |  |  |  |
| --- | --- | --- | --- |
| Milestone or deliverable | Duration  (hours or days) | Projected start date | Anticipated end date |
| Trained ML Model | 2 days | 02/24/2024 | 02/26/2024 |
| CLI Application | 1 day | 02/27/2024 | 02/28/2024 |
| User Guide | 2 hours | 02/29/2024 | 02/29/2024 |

## Evaluation Plan

* Describe the verification method(s) to be used at each stage of development.
  + Phase 1: Pre-Development
    - Requirements Validation: Confirm that the project objectives, scope, and machine learning solution requirements accurately reflect stakeholder needs and expectations.
    - Data Readiness Assessment: Ensure the collected dataset is of high quality, diverse, and sufficient for training the machine learning model. This includes checking for biases, missing values, and data diversity.
  + Phase 2: Development
    - Model Prototyping: Evaluate the feasibility of different machine learning algorithms and select the most promising approach based on performance metrics such as accuracy and processing time.
    - Model Development and Training: Verify the model's accuracy, generalization ability, and robustness across different subsets of the data to prevent overfitting and underfitting.
    - Integration With CLI Application: Ensure the machine learning model's API interfaces correctly with the Translatinator app.
  + Phase 3: Testing
    - Testing: Validate individual components (unit testing), their interactions (integration testing), and the overall solution (UAT) to ensure it meets user expectations and requirements.
    - Assess and enhance the machine learning model and app feature's performance, based on quantitative benchmarks and qualitative user feedback.
  + Phase 4: Deployment and Monitoring
    - Preparation: Ensure all elements of the feature are ready for launch, including final testing of the deployment environment and infrastructure.
    - Launch: Immediately identify and address any issues upon release, monitoring system performance and user engagement in real time.
    - Post-Launch Support: Ongoing evaluation of the feature's performance and user satisfaction to inform future improvements and adjustments.
  + Phase 5: Improvements
    - A/B Testing: Test new updates or features against the current version to evaluate improvements or changes in user engagement and satisfaction, guiding future development priorities.
* Describe the validation method to be used upon completion of the project.
  + Technical Accuracy Validation
    - Model Accuracy Metrics: Utilize standard machine learning metrics such as precision, recall, F1 score, and accuracy to evaluate the model's performance on a validation dataset. These metrics will provide insights into how well the model identifies cats and dogs, balancing between correctly labeling the objects and minimizing false positives or negatives.
    - Confusion Matrix Analysis: Use a confusion matrix to visualize the model's predictions versus the actual labels, identifying any patterns of misclassification. This analysis is particularly useful for determining if the model struggles with specific breeds or lighting conditions.
  + User Experience Validation
    - User Acceptance Testing (UAT): Conduct UAT with a diverse group of users to assess the feature's usability, intuitiveness, and overall satisfaction. Participants will be asked to use the object recognition feature under various conditions and provide feedback on their experience.
    - Beta Testing: Release the feature to a limited user base to gather real-world usage data and feedback. Beta testing will help identify any unforeseen issues or usability challenges not captured during UAT.
    - Surveys and Feedback Forms: Distribute surveys and feedback forms to both UAT participants and beta testers to collect quantitative and qualitative feedback on the feature's performance, user interface design, and overall impact on the language learning experience.
  + Combined Validation Approaches
    - A/B Testing: If feasible, conduct A/B testing with the new feature, presenting it to one group of users while withholding it from another. This method can validate the feature's effectiveness in enhancing the language learning experience by comparing engagement metrics, retention rates, and learning outcomes between the two groups.
    - Analytics and Usage Metrics: Analyze usage metrics such as frequency of feature use, average session length, and progression in learning objectives when the feature is used. These metrics will validate the feature's impact on user engagement and learning efficiency.
    - Continuous Monitoring and Feedback Loop: After the feature's public release, continue to monitor user feedback and performance metrics. This ongoing validation process ensures that any issues are promptly addressed and that the feature continues to meet user needs and expectations.

## Resources and Costs

* Developer Costs
  + Developer time: 200 hours at $40/hour = $8,000
* Project Management Costs
  + Project Manager: 100 hours at $30/hour = $3,000
* Hardware Costs
  + Servers to train and store the model: $500/month
* Software Costs
  + Software (TensorFlow): Free
* Total Project Costs:
  + Developer Costs: $8,000
  + Project Management Costs: $3,000
  + Hardware Costs: $500/month (for the duration of the project and ongoing maintenance)
  + Software Costs: $0 (TensorFlow is free)

# Part C: Application

Part C is your submitted application. This part of the document can be left blank or used to include a list of any submitted files or links.

The minimal requirements of the submitted *application* are as follows:

1. **The application functions as described.** Following the ‘User Guide’ in part D, the evaluator must be able to successfully review your application on a Windows 10 machine.
2. **A mathematical algorithm applied to data,** e.g, supervised, unsupervised, or reinforced machine learning method.
3. **A “user interface.”** Following the ‘User Guide’ in part D, the client must be able to use the application towards solving the proposed problem (as described in parts A, B, and D). For example, the client can input variables and the application outputs a prediction.
4. **Three visualizations.** The visualizations can be included separately when including them in the application is not ideal or possible, e.g., the visualizations describe proprietary data but the application is customer-facing.
5. **Submitted files and links are static and accessible.** All data, source code, and links must be accessible to evaluators on a Windows 10 machine. If parts of the project can be modified after submission, then matching source files must be submitted. For example, if the application is a website or hosted notebook, the .html or .ipynb files must be submitted directly to assessments.

Ideally, submitted applications should be reviewable using either Windows or Mac OS, e.g., Jupyter notebooks, webpages, Python projects, etc. If the source files exceed the 200 MB limit, consider providing screenshots or a Panopto video of the functioning application and contact your course instructor.

# Part D: Post-implementation Report

Create a post-implementation as outlined below. Provide sufficient detail so that a reader knowledgeable in computer science but unfamiliar with your project can understand what you have accomplished. Using examples and visualizations (including screenshots) beyond the three required is recommended (but not required). **Write everything in the past tense.**

## Solution Summary

* Summarize the problem and solution.
* Describe how the application provides a solution to the problem from parts A and B.

## Data Summary

* Provide the source of the raw data, how the data was collected, or how it was simulated.
* Describe how data was processed and managed throughout the application development life cycle: design, development, maintenance, or others.

## Machine Learning

For each employed method (at least one is required) provide the following:

* Indentify the method and what it does (the “what”).
* Descirbe how the method was developed (the “how”).
* Justifiy the selection and developement of the method (the “why”).

## Validation

For each employed method described in the section above provide the following:

* An appropiate validaiton method.An appropiate validaiton method.
* Results of the validation method or a future plan to obtain those results.

## Visualizations

Identify the location of at least three unique visulizations. They can additionally included here.

## User Guide

Include an enumerated (steps 1, 2, 3, etc.) guide to execute and use your application.

* Include instructions for downloading and installing any necessary software or libraries.
* Provide an example of how the client should use the application.

# Reference Page

Following APA guidelines, include references for any cited works, e.g., (Author, year). References are not requried, and this page can be removed if no references are used. To cite sources used for code, you should include the referfences as code comments within the source code.