

Development Plan CXR

Team 25, Neuralyzers
Ayman Akhras
Nathan Luong
Patrick Zhou
Kelly Deng
Reza Jodeiri

Table 1: Revision History

Date	Developer (s)	Change
Date1	Name (s)	Description of changes
Date2	Name (s)	Description of changes
...

[Put your introductory blurb here. Often the blurb is a brief roadmap of what is contained in the report. —SS]

[Additional information on the development plan can be found in the lecture slides. —SS]

1 Confidential Information?

There is no confidential information to protect.

2 IP to Protect

[State whether there is IP to protect. If there is, point to the agreement. All students who are working on a project that requires an IP agreement are also required to sign the “Intellectual Property Guide Acknowledgement.” —SS]

3 Copyright License

Our Team is adopting an MIT License, which is a permissive open-source license. This license allows users to freely use, copy, modify, merge, publish, distribute, sublicense, and even sell copies of the software, provided that the original copyright notice and the license text are included with all copies or substantial portions of the software. The license can be found in our github repository at the

- Link: <https://github.com/RezaJodeiri/CXR-Capstone/blob/main/LICENSE>

4 Team Meeting Plan

Our team plans to meet twice a week on Discord (virtual meetings) to discuss upcoming deliverables and development progress on AI models. OUR team plans to meet with the supervisor at least once a month via Teams. In the meetings our team plans to update the supervisor about our current progress, what challenges we are facing, what are the next steps and what advice does the instructor has.

5 Team Communication Plan

Neuralanalyzers plans to hold in person and virutual meetings as a method of communication between the team.

5.1 Discord

Discord will be used as the team’s main mehtod of communication as its great at message logging, file sharing, creating threads for issues, real time communication and creating channels to separate information. Discord will be used by team members to update, and send key information or documentation links such as GitHub links, YouTube links, and code blocks to help keep the team up to date as well as ensure note taking is kept at a high standard.

5.2 GitHub

GitHub is a great resource that Neuralanalyzers will be utilizing to create branches and track code whilst ensuring clean merging of code between members. On github, issue requests can be created for the team to be notified and add input, to fix following issues as well as update the project as a whole. This will be the key center point where all the documentation and code will be saved.

6 Team Member Roles

- Nathan Luong
 - Scrum Master

- Developer
- Machine Learning Expert: Will do research on how to effectively read the X-Ray Image and create models that can be used to read race, age, and details of the patient.
- Ayman
 - Developer
 - Note Taker
 - Computer Vision Expert: Will do research on how using Pytorch or Cuda could be utilized to ensure best performance of Application.
- Patrick Zhou
 - Developer
 - Reviewer
 - Python Expert: Will be in charge of Documentation of Python code Labeling each Function as well as, keeping the programming style consistent. Will be able to transform our pseudocode into Python code.
- Kelly Deng
 - Developer
 - Meeting Chair
 - Machine Learning Expert: Will do research on how to effectively read the X-Ray Image and create models that can be used to read race, age, and details of the patient.
- Reza Jodeiri
 - Developer
 - Leader
 - Chest X-ray Expert: Will lead the research and guide the team towards achieving the project's objectives through their deep understanding of chest X-rays. Their expertise will be instrumental in ensuring that the X-ray images are accurately interpreted, identifying key anomalies and common disease patterns. This knowledge will directly contribute to training the machine learning model by selecting the most relevant features and markers for disease detection, such as abnormalities in lung structure, nodules, or lesions.

7 Workflow Plan

- Usage of Git, Github
 - The capstone repository (<https://github.com/RezaJodeiri/CXR-Capstone>) will be actively maintained by all 5 members.
 - Many Core Github features will be used extensively in order to drive our capstone project.
 - * **Github Issues:** to track Issues, Tickets, Bugs, and Tasks
 - * **Github Repository:** to monitor, and track changes, and store source code of the capstone project.
 - * **Github Actions:** to run automation for continuous integration tasks. For example: Unit tests or linting check per pull requests.
 - * **Github Branch Protection:** to protect developers from accidental code push onto main branch.
 - * **Github Approval Rules:** to make sure that changes are approved by all 4 other team members, before being merged into main.
- Pull Requests
 - All Pull Requests will come with a template. This template can be found under `.github/pull_request_template`

- On every pull requests, members are required to provide detail descriptions, and checklist, ensuring standardized code-review process.
- Branching Strategy
 - Branching Strategy might differ from case-to-case. However, general development guideline will encourage members to follow a trunk-based development approach.
 - Each members will be responsible to keep their local repos up-to-date with the remote repo, then each member will work on the ticket/issue/task, and open a pull request against the main branch.
 - With this branching strategy, each team members would need to perform rebasing on a regular basics, to avoid big merge conflicts, and potential efficiency lost.
 - The branch name should be descriptive of the change that being developed. For example: **dev_plan_MEMBER_NAME** or **feature/training-data-labelling**
- Issues/ Tickets Tracking & Management
 - Issues tracking and management will be conducted via Github Issues. We chose this platform instead of Jira or Trello, since GitHub can be a one-stop-shop, and preventing context switching for team members.
 - Each issue must contain a tag (ie documentation, bug, duplicate, feature-request) to classify the issue type.
 - Each issue must contain an assignee, and the assignee is responsible to investigate and close the issue.
- Usage of Continuous Integration / Continuous Deployment
 - As mentioned, Github Actions will be used for CI/CD automation.
 - CI/CD can be utilized for unit testing, linting, security checks, automatic deployment onto cloud environment (AWS, GCP).
 - CI/CD automation would need to be developed in conjunction with feature development in order to reduce human error, and repetitive work.

8 Project Decomposition and Scheduling

[How will the project be scheduled? This is the big picture schedule, not details. You will need to reproduce information that is in the course outline for deadlines. —SS]

- How will you be using GitHub projects?
 - We will be using GitHub projects to keep track of the progress of the project. We will create a project board with columns such as To Do, In Progress, and Done. Each task will be represented as an issue on the project board. The project board will be updated regularly to reflect the current status of the project.
- Include a link to your GitHub project
 - <https://github.com/RezaJodeiri/CXR-Capstone>

9 Proof of Concept Demonstration Plan

10 Expected Technology

Component	Technology / Description
Programming Language	<ul style="list-style-type: none"> • Python: Chosen for its extensive support for machine learning and medical imaging libraries and models. • JavaScript: For developing the web-based user interface and interactive visualization components that allows users to upload chest X-ray images and view the results of the AI model. • Groovy/Bash: For writing scripts to automate the deployment of the application and setting up CI/CD pipelines.
Machine Learning Framework	PyTorch: Selected due to its flexibility, dynamic computation graphs, and support for GPU acceleration, which are essential for training deep learning models on large datasets.
Medical Imaging Libraries	<ul style="list-style-type: none"> • TorchXRayVision: To leverage pre-trained models and Chest X-ray datasets such as CheXpert and tools specifically designed for medical image analysis, particularly chest X-rays. • scikit-learn: For additional image preprocessing tasks like resizing and normalization.
Pre-trained Models	ResNet: Utilized as a backbone architecture for transfer learning to classify chest X-ray images. Pre-trained models will be fine-tuned on the project's dataset(Imagenome) to yield desired accuracy and performance.
Linter Tool	Pylint: To ensure code adheres to Python's coding standards and improve code quality through automated static analysis.
Unit Testing Framework	Pytest: For writing and automating tests to verify the reliability and functionality of machine learning models and components.
Code Coverage Tool	Coverage.py: Integrated with Pytest to measure the percentage of code covered by unit tests, ensuring robust testing of the codebase.
Continuous Integration	GitHub Actions: Automates testing, linting, and other checks when code is pushed to the repository.
Development Environment	<ul style="list-style-type: none"> • Visual Studio Code: Chosen for its extensive Python support, ease of use, and integration with GitHub for version control and collaboration. • Docker: To containerize our packages and dependencies, ensuring consistency across different environments and simplifying deployment.
Documentation Tool	LaTeX/Overleaf: For writing and formatting our capstone documents, reports, and deliverables.

Table 2: Technology Choices for Capstone Project

11 Coding Standard

[What coding standard will you adopt? —SS]

Appendix — Reflection

[Not required for CAS 741 —SS]

1. Why is it important to create a development plan prior to starting the project?
 - It is crucial to create a development plan prior to starting the project as it aims to ensure that the project is completed on time and within budget. The development plan outlines the tasks that need to be completed, the resources that are required, and the timeline for completion. By creating a development plan, the team can identify potential risks and issues early on and develop strategies to mitigate them. The development plan also helps to ensure that all team members are on the same page and working towards the same goals.
2. In your opinion, what are the advantages and disadvantages of using CI/CD?

Advantages of using CI/CD include:

- Improved code quality: CI/CD runs automated tests on the code to identify bugs and issues early on, which can help to improve the overall quality of the code.
 - Faster Development: By using CI/CD we can encourage/create faster development cycles that automate the process of building, testing, and deploying code, which can help to speed up the development process.
 - Reduced Human Error: By making the deployment process automated, the chance manual errors when deploying is significantly reduced, ensuring reliability and consistency in releases.
3. What disagreements did your group have in this deliverable, if any, and how did you resolve them?
 - Our group did not have any disagreements in this deliverable.

Appendix — Team Charter

[borrows from University of Portland Team Charter —SS]

External Goals

[What are your team’s external goals for this project? These are not the goals related to the functionality or quality fo the project. These are the goals on what the team wishes to achieve with the project. Potential goals are to win a prize at the Capstone EXPO, or to have something to talk about in interviews, or to get an A+, etc. —SS]

Attendance

Expectations

[What are your team’s expectations regarding meeting attendance (being on time, leaving early, missing meetings, etc.)? —SS]

Acceptable Excuse

[What constitutes an acceptable excuse for missing a meeting or a deadline? What types of excuses will not be considered acceptable? —SS]

In Case of Emergency

[What process will team members follow if they have an emergency and cannot attend a team meeting or complete their individual work promised for a team deliverable? —SS]

Accountability and Teamwork

Quality

[What are your team’s expectations regarding the quality of team members’ preparation for team meetings and the quality of the deliverables that members bring to the team? —SS]

Attitude

[What are your team’s expectations regarding team members’ ideas, interactions with the team, cooperation, attitudes, and anything else regarding team member contributions? Do you want to introduce a code of conduct? Do you want a conflict resolution plan? Can adopt existing codes of conduct. —SS]

Stay on Track

[What methods will be used to keep the team on track? How will your team ensure that members contribute as expected to the team and that the team performs as expected? How will your team reward members who do well and manage members whose performance is below expectations? What are the consequences for someone not contributing their fair share? —SS]

[You may wish to use the project management metrics collected for the TA and instructor for this. —SS]

[You can set target metrics for attendance, commits, etc. What are the consequences if someone doesn’t hit their targets? Do they need to bring the coffee to the next team meeting? Does the team need to make an appointment with their TA, or the instructor? Are there incentives for reaching targets early? —SS]

Team Building

[How will you build team cohesion (fun time, group rituals, etc.)? —SS]

Decision Making

[How will you make decisions in your group? Consensus? Vote? How will you handle disagreements? —SS]