

Software Engineering Department  
ORT Braude College

Capstone Project Phase A – 61998

**Strep Throat detection using Machine Learning**

**24-2-D-22**

**Dr. Natali Levi**

**Matan Nachmuha** – [Matan.Nachmuha@e.braude.ac.il](mailto:Matan.Nachmuha@e.braude.ac.il)

**Yehuda Bitton** – [Yehuda.Bitton@e.braude.ac.il](mailto:Yehuda.Bitton@e.braude.ac.il)

**GitHub** - https://github.com/nchmoka/Capstone

**Table of Contents**

1. Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
2. Related Work . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
3. Background. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5

3.1 Strep Throat . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. . . . . . . . 5

3.2 Machine Learning and Image Processing . . . . . . . . . . . . . . . .. . . . . . . . .6

3.3 Application of Machine Learning in Strep Throat Diagnosis .. . . . . . . . . 7

3.4 Mobile Technology and Accessibility . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

3.5 Data Privacy and Security. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

1. Expected Achievements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

4.1 Outcomes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

4.2 Unique Features . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10

4.3 Criteria for Success . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11

1. Research . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12

5.1 Research - Strep Throat . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .12

5.2 Research - Machine Learning and Image Processing . . . . . . . . . . . . . . . 13

5.3 Development of the Diagnostic Tool. . . . . . . . . . . . . . . . . . . . . . . . . . . . 14

5.4 Continuous Learning and Improvement . . . . . . . . . . . . . . . . . . . . . . . . .15

1. Product . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16

6.1Requirements. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16

6.2 System Architecture . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 19

6.3 User Interface and Experience . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 21

6.4 Testing and Quality Assurance . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 21

6.5 Continuous Improvement and Updates . . . . . . . . . . . . . . . . . . . . . . . . . . 21

6.6 Diagrams . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22

1. Verification and Evaluation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .24

7.1 Evaluation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24

7.2 Verification . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 26

8. References . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .29

**Abstract**

Strep throat, caused by Group A Streptococcus (GAS), requires timely diagnosis to prevent severe complications. Traditional methods, such as throat swab cultures and rapid antigen detection tests (RADTs), are often time-consuming and require laboratory facilities. This project aims to develop a machine learning-based screening tool for strep throat that leverages image processing and predictive modeling to provide a fast, accurate, and accessible diagnostic alternative. The proposed system will analyze throat images captured using a camera and classify them based on the presence of strep throat, offering preliminary diagnosis to guide further medical consultation.

The mobile application will use a convolutional neural network (CNN) trained on labeled throat images to identify visual markers indicative of strep throat. Users can capture throat images with a smartphone, which the app will preprocess and analyze to provide immediate diagnostic feedback. The project aims to democratize access to strep throat diagnosis, making it available to people in remote or underserved areas and reducing the burden on healthcare facilities.

Overall, this project represents a significant step forward in integrating machine learning with healthcare, offering a practical solution to improve patient outcomes and advance medical diagnostics.

1. **Introduction**

Strep throat is an infection in the throat and tonsils caused by Group A Streptococcus bacterium [1][2]. It is particularly common in children in the ages of 7-8 [3] but can affect people of all ages. The diagnosis is typically based on clinical symptoms [4], including a sore throat, fever, and swollen lymph nodes, and confirmed through a throat swab culture or a rapid antigen detection test (RADT). However, these methods can be slow and require specialized equipment and trained personnel, leading to delays in diagnosis and treatment. Early detection and treatment are crucial to prevent complications [4] such as rheumatic fever, peritonsillar abscess, and other severe conditions that can arise from untreated strep throat.

The advent of machine learning and advanced image processing techniques offers a promising alternative for rapid and non-invasive diagnosis. By analyzing images of the throat, a machine learning model can identify patterns and markers indicative of strep throat, providing an immediate and accessible diagnostic tool. This approach not only speeds up the diagnostic process but also makes it more accessible to people in remote or underserved areas where laboratory facilities may be limited.

This project proposes the development of a mobile application that utilizes machine learning algorithms to analyze throat images and provide a preliminary diagnosis of strep throat. The application aims to be user-friendly, requiring only a smartphone camera and an internet connection, making it a valuable tool for both healthcare professionals and the public.

The mobile application will allow users to take a photo of their throat using their smartphone’s camera, which will then be analyzed by the machine learning model. The image will be processed in real-time, with the application evaluating visual markers like redness, swelling, or white spots on the tonsils—common indicators of strep throat. This immediate analysis will help individuals quickly determine whether they need further medical intervention.

This technology is highly beneficial because it provides a quick, non-invasive, and accessible way for people to check for strep throat, especially in areas where healthcare services may be hard to reach. It reduces the need for in-person appointments, saving both time and resources while ensuring that individuals can get faster diagnosis and treatment. Moreover, it empowers users to take proactive steps in managing their health, which can lead to better health outcomes and reduced risk of complications.

**2. Related Work**

In recent years, the application of machine learning (ML) in medical diagnostics has gained significant momentum, offering promising solutions for more accurate and accessible diagnosis of various diseases. One area that has benefited from such advancements is the diagnosis of strep throat, a common bacterial infection caused by *Streptococcus pyogenes*. Traditional methods for diagnosing strep throat include the Centor score, throat cultures, and rapid antigen detection tests (RADTs). However, these methods can sometimes be slow, costly, or limited in accuracy [5][6].

To address these challenges, researchers have developed novel approaches leveraging machine learning and image processing techniques for strep throat detection using smartphone-based applications. One such method [6] utilized a smartphone's built-in camera to capture throat images, which were then analyzed using color correction and image segmentation algorithms to identify features indicative of streptococcal pharyngitis. The study employed a k-nearest neighbor (k-NN) classifier to distinguish between healthy and infected throats, achieving an accuracy of 93.75% [6].

Similarly, Tae Keun Yoo et al. (2020) explored the use of deep learning models for automated detection of severe pharyngitis from throat images taken by smartphones. Their study utilized a convolutional neural network (CNN) architecture, particularly ResNet50, combined with generative adversarial networks (GANs) for data augmentation. The model demonstrated impressive performance, with a detection accuracy of 95.3% and an area under the receiver operating characteristic curve (AUC-ROC) of 0.988, making it highly effective for real-time screening of pharyngitis [5].

Additionally, [7] focused on bacterial image analysis using deep learning approaches for clinical microscopy, emphasizing the need for automated detection and classification of bacteria. Their study utilized multiple deep learning models, including YOLOv4, to classify different stages of bacterial growth from microscopic images, achieving a mean average precision (mAP) of 98% [7]. The success of these models highlights the potential of deep learning in medical diagnostics, particularly for identifying bacterial infections like strep throat.

The integration of machine learning with smartphone technology presents an exciting frontier for telemedicine, allowing for timely and accurate diagnosis of diseases such as strep throat. As these technologies continue to evolve, they hold the potential to reduce the burden on healthcare systems by providing accessible diagnostic tools for remote and underserved populations [5][7].

**3. Background**

3.1 Strep Throat

Strep throat is an infection in the throat and tonsils caused by the bacterium Streptococcus pyogenes, also known as Group A Streptococcus (GAS). This condition is particularly common among children but can affect individuals of all ages. Strep throat is highly contagious, spreading through respiratory droplets when an infected person coughs or sneezes. It can also spread through shared food or drinks, and close contact with infected individuals.

Symptoms and Diagnosis

Common symptoms of strep throat include:

* Sore Throat: Pain that typically starts quickly and can be severe.
* Red and Swollen Tonsils: Often with white patches or streaks of pus.
* Tiny Red Spots: Located on the roof of the mouth.
* Swollen Lymph Nodes: Particularly in the neck.
* Headache: Accompanied by throat pain.
* Nausea and Vomiting: Especially in younger children.
* Rash: Known as scarlet fever, which is a sign of a more severe infection.
* Fever

[4]

The diagnosis of strep throat is typically confirmed through two main methods:

1. Rapid Antigen Detection Test (RADT): A quick test that can provide results within minutes by detecting antigens in the throat swab. It has high specificity but moderate sensitivity, meaning it may miss some cases. [1][8]
2. Throat Culture: A more accurate test that involves culturing the bacteria from a throat swab. It takes 24-48 hours to provide results but is considered the gold standard for diagnosis. [1][8]

Prompt diagnosis and treatment are essential to prevent complications such as:

* Rheumatic Fever: An inflammatory disease that can affect the heart, joints, skin, and brain.
* Post-Streptococcal Glomerulonephritis: A kidney disease that can develop after strep throat.
* Peritonsillar Abscess: A collection of pus behind the tonsils that can cause severe pain and difficulty swallowing.

[4]

3.2 Machine Learning and Image Processing

Machine learning, particularly deep learning, has revolutionized many fields, including medical imaging. Deep learning models, such as convolutional neural networks (CNNs), have shown remarkable success in image classification tasks. These models can automatically learn and extract features from raw image data, enabling accurate classification without the need for manual feature engineering.

Convolutional Neural Networks (CNNs)

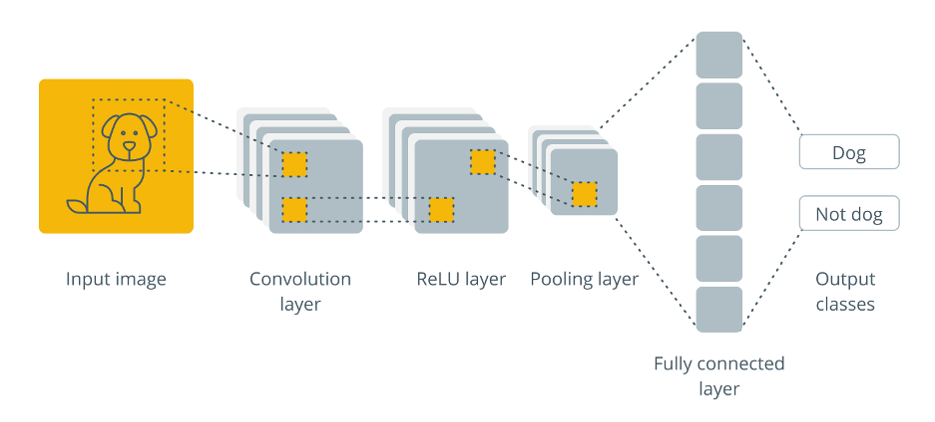


Figure 1: Convolutional neural network

CNNs are a type of deep learning model designed specifically for image processing tasks. They consist of multiple layers, including convolutional layers, pooling layers, and fully connected layers:

Convolutional Layers: These layers apply convolution operations to the input images using filters (or kernels) to extract features such as edges, textures, and patterns.

Pooling Layers: These layers reduce the spatial dimensions of the feature maps, which helps in reducing the computational load and makes the model more robust to variations in the input images.

Fully Connected Layers: These layers are like those in traditional neural networks and are used to combine the features extracted by the convolutional layers to make final predictions.

CNNs have been used successfully in various medical imaging applications, including:

Skin Cancer Detection: CNNs have been trained to classify skin lesions as benign or malignant with high accuracy.[9]

Pneumonia Detection: CNNs have been used to analyze chest X-rays and detect pneumonia, often performing at a level comparable to radiologists.[10]

Diabetic Retinopathy: CNNs have been applied to retinal images to detect signs of diabetic retinopathy, a common complication of diabetes.[11]

Data Augmentation

To improve the performance and robustness of CNN models, data augmentation techniques are often used. Data augmentation involves creating modified versions of the original images through transformations such as:

Rotation: Rotating images by a certain degree to make the model invariant to orientation changes.

Scaling: Zooming in or out of images to make the model robust to size variations.

Flipping: Flipping images horizontally or vertically to increase diversity in the training data.

Brightness and Contrast Adjustment: Modifying the brightness and contrast of images to make the model robust to different lighting conditions.

Preprocessing

Image preprocessing is a crucial step in preparing the data for training a CNN model. Common preprocessing steps include:

Normalization: Scaling the pixel values to a standard range (e.g., 0 to 1) to ensure consistent input to the model.

Resizing: Adjusting the image dimensions to match the input size expected by the CNN model.

Cropping: Removing irrelevant parts of the image to focus on the region of interest.

3.3 Application of Machine Learning in Strep Throat Diagnosis

Applying machine learning to strep throat diagnosis involves several key steps:

Data Collection: Gathering a large dataset of labeled throat images. These images should represent various conditions, including healthy throats, strep throat, and other throat infections.

Model Training: Training a CNN model on the labeled dataset to learn the features associated with strep throat. This involves splitting the dataset into training, validation, and test sets of 10 percent from the dataset to evaluate the model's performance.

Evaluation: Assessing the model's performance using metrics such as accuracy, precision, recall, and F1-score. These metrics help determine how well the model can distinguish between strep throat and other conditions.

Deployment: Integrating the trained model into a mobile application that can capture throat images using a smartphone camera and provide real-time diagnostic feedback.

3.4 Mobile Technology and Accessibility

Recent advancements in smartphone technology have made it possible to capture high-quality medical images using mobile devices. This has opened new possibilities for developing mobile applications that can assist in the diagnosis of various conditions. For example, mobile applications for skin cancer detection and diabetic retinopathy screening have demonstrated the potential of combining machine learning with mobile technology to provide accessible and accurate diagnostic tools.

For strep throat diagnosis, a mobile application can leverage the smartphone camera to capture throat images. The application can guide users through the process of capturing high-quality images, preprocess these images to enhance their quality, and use a trained CNN model to analyze the images and provide a preliminary diagnosis. This approach makes strep throat diagnosis more accessible, especially for individuals in remote or underserved areas where laboratory facilities may be limited.

תמונה שמכילה פני אדם, טקסט, ציור, אומנות קליפיפם

התיאור נוצר באופן אוטומטי

Figure 2: User takes picture of his throat

3.5 Data Privacy and Security

Ensuring the privacy and security of user data is a critical consideration in developing any medical application. The mobile application for strep throat diagnosis will incorporate several measures to protect user data:

Encryption: All data transmitted between the user's device and the server will be encrypted using industry-standard encryption protocols.

Secure Storage: User data will be stored in secure databases with access controls to prevent unauthorized access.

Anonymization: Personal identifiers will be removed from the data to protect user privacy.

**4. Expected Achievements**

4.1 Outcomes

The outcomes we aim to achieve with this project center on creating a practical and accessible diagnostic tool for strep throat detection. Our goal is to provide a user-friendly mobile application that can be utilized by both healthcare professionals and the public. The tool will analyze throat images captured using a smartphone and output a preliminary diagnosis of strep throat, indicating whether the user should seek further medical consultation.

The application will utilize a convolutional neural network (CNN) to identify signs of strep throat based on visual markers present in the images. This data-driven approach will allow us to draw insights about the presence of the infection, with the potential to enhance early detection and treatment.

4.2 Unique Features

4.2.1 Mobile Application Platform

The mobile application is designed to be a unique feature, leveraging the widespread availability of smartphones. A key challenge will be ensuring the app functions seamlessly across different devices and operating systems. This will involve understanding the capabilities and limitations of mobile hardware to capture high-quality images and process them effectively.

4.2.2 CNN-Based Image Analysis

Implementing a CNN for image analysis is critical for the app's diagnostic capabilities. The model must be robust enough to differentiate between strep throat and other throat conditions under varying lighting conditions and image qualities. This involves training the model on a diverse dataset to ensure high accuracy and reliability.

4.2.3 User Experience and Guidance

A significant aspect of the application will be the user interface (UI) and experience (UX). The app will need to provide clear instructions for users to capture suitable throat images, reducing the potential for user error. The design will emphasize ease of use, ensuring that even individuals without technical expertise can navigate the app and understand the results.

4.2.4 Data Security and Compliance

Given the sensitivity of health data, the application will incorporate stringent data security measures. This includes encryption of personal information, secure storage of diagnostic results.

4.2.5 Continuous Model Improvement

The application will feature mechanisms for continuous improvement of the CNN model. As new data is collected, the model will be periodically retrained to enhance its accuracy. This adaptive learning capability is crucial for maintaining the relevance and accuracy of the diagnostic tool over time.

4.2.6 Educational Component

The app will also include an educational component, offering information about strep throat, its symptoms, and the importance of medical treatment. This feature aims to inform users and encourage proactive health management.

4.3 Criteria for Success

Diagnostic Accuracy: The final product should provide accurate diagnostic feedback for strep throat. To ensure this, we will use a dataset of diagnosed images (data) and run on our model to check whether the diagnosis provided by our application matches the actual diagnosis. Performance metrics such as accuracy, precision, and recall should be comparable to traditional diagnostic methods.

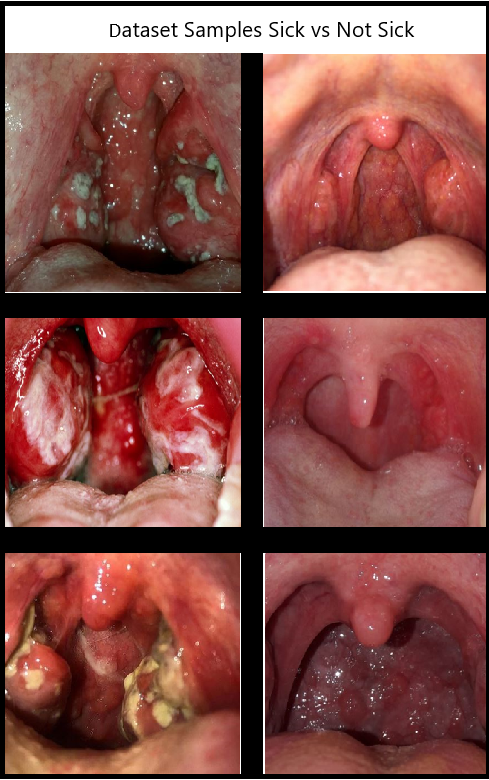


Figure 3: dataset Samples

User Accessibility: The application must be intuitive and accessible, with a straightforward user interface that guides users through the image capture process and clearly presents the diagnostic results.

Technical Performance: The app should function smoothly, providing real-time diagnostic feedback without significant delays. The CNN model should be optimized for efficiency, allowing quick processing of images.

Security and Compliance: The app must ensure the security and privacy of user data, while maintaining user trust.

Continuous Improvement: The system should have a robust mechanism for model updates and improvement, ensuring that the diagnostic tool remains accurate and relevant with the latest medical knowledge and data.

Impact and Utility: The application should effectively contribute to the early detection and management of strep throat, potentially reducing the burden on healthcare systems by providing a reliable preliminary diagnosis tool accessible to a broad audience.

By meeting these criteria, the project aims to deliver a reliable, user-friendly, and secure diagnostic tool for strep throat, enhancing healthcare accessibility and effectiveness.

5. **Research**

5.1 Research – Strep Throat

Strep throat is a bacterial infection caused by Group A Streptococcus (GAS) [1][2], primarily affecting the throat and tonsils [4]. It spreads through respiratory droplets [3] and is most common in children at ages of 7-9 [3] but can affect individuals of all ages. Key symptoms include sore throat, red and swollen tonsils, fever, and swollen lymph nodes [4]. If left untreated, strep throat can lead to serious complications such as rheumatic fever, kidney inflammation, and peritonsillar abscesses [4].

5.1.1 Transmission and Epidemiology

Strep throat spreads easily in environments with close contact, such as schools and households [3]. It is most prevalent in winter, and early spring [1]. Understanding the transmission and epidemiology of strep throat is essential for developing effective diagnostic tools.

5.1.2 Current Diagnostic Methods

Diagnosis is typically confirmed through:

Rapid Antigen Detection Test (RADT): Provides quick results but may miss some cases due to lower sensitivity. [1][8]

Throat Culture: The gold standard, offering high accuracy but requiring 24-48 hours for results. [1][8]

While effective, these methods have limitations, particularly in terms of accessibility and speed, highlighting the need for a faster, more accessible diagnostic tool.

5.2 Research – Machine Learning and Image Processing

Machine learning, particularly convolutional neural networks (CNNs), has proven highly effective in medical image analysis [7]. CNNs can automatically learn features from images, making them ideal for diagnosing conditions like strep throat through throat image analysis.

5.2.1 Convolutional Neural Networks (CNNs)

CNNs are composed of convolutional layers, pooling layers, and fully connected layers that work together to extract and analyze features from images. They are widely used in medical diagnostics due to their ability to handle complex image data and make accurate predictions.

תמונה שמכילה טקסט, צילום מסך

התיאור נוצר באופן אוטומטי

Figure 4: Detection of strep throat model

5.2.2 Data Augmentation

To improve the model's performance, data augmentation techniques such as rotation, scaling, and brightness adjustment will be used. These techniques help create a more diverse training dataset, reducing the risk of overfitting and improving the model's generalization to new images.

5.2.3 Image Preprocessing

Image preprocessing is critical to ensuring the quality of input data. Steps include normalization, resizing, noise reduction, and contrast enhancement. These processes prepare the images for effective analysis by the CNN, ensuring that the model receives consistent and clear inputs.

5.3 Development of the Diagnostic Tool

The diagnostic tool will be developed as a mobile application with several key components:

5.3.1 Dataset Collection

A diverse dataset of throat images, including healthy throats, strep throat cases, and other conditions, will be collected. This dataset will be essential for training the CNN model to distinguish between different conditions accurately.

5.3.2 Model Training and Validation

The CNN model will be trained on the collected dataset, with a portion reserved for validation and testing. The training process will involve optimizing hyperparameters and using techniques like regularization to prevent overfitting. The model's performance will be evaluated using metrics such as accuracy, precision, recall, and F1-score.

5.3.3 Integration with Mobile Application

The trained model will be integrated into a mobile app that guides users through image capture and provides real-time diagnostic feedback. The app will include features to ensure high-quality image capture and secure data storage.

5.4 Continuous Learning and Improvement

The diagnostic tool will include mechanisms for continuous learning. New data collected from users will be used to retrain the model periodically, ensuring it remains accurate and up to date. User feedback will also be incorporated to improve the app's performance and usability.

This research phase lays the foundation for developing a reliable, user-friendly diagnostic tool that leverages machine learning to provide fast, accurate strep throat diagnosis, ultimately improving healthcare accessibility and outcomes.

**6. Product**

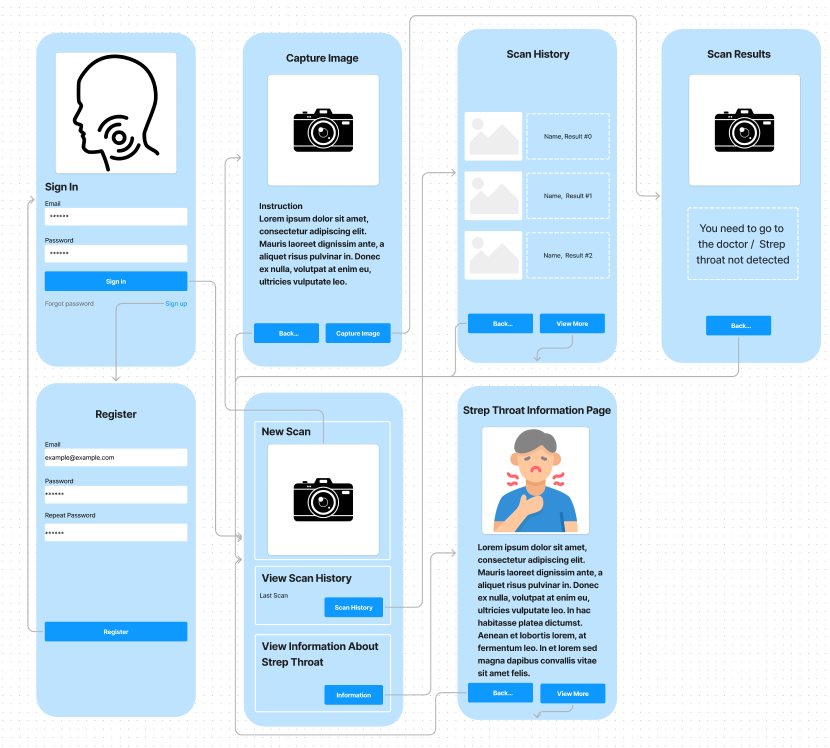


Figure 5: App Screens

6.1 Requirements

The product is a mobile application designed to diagnose strep throat using machine learning. It will combine advanced image processing, a robust backend infrastructure, and a user-friendly interface to deliver accurate diagnostic feedback. The requirements for the product are categorized into functional and non-functional specifications.

6.1.1 Functional Requirements

|  |  |
| --- | --- |
| 1 | The application shall enable users to capture high-quality images of their throats using the smartphone camera. |
| 2 | The app will provide real-time guidance, ensuring that users position the camera correctly and capture images that meet the quality standards necessary for accurate analysis. |
| 3 | The system shall preprocess the captured images to enhance their quality. This includes normalization, noise reduction, and contrast enhancement to ensure consistency and clarity. |
| 4 | The app shall use a convolutional neural network (CNN) model to analyze the preprocessed images and determine the likelihood of strep throat. |
| 5 | The model’s output will be a probability score indicating the presence of strep throat, accompanied by a recommendation for further medical consultation if necessary. |
| 6 | The application shall provide immediate diagnostic feedback to the user after the image is analyzed. The feedback will include the probability of strep throat and suggested next steps. |
| 7 | The app shall securely store user diagnostic results in a database, allowing users to access their past diagnoses. |
| 8 | The system will include options for users to update their profile, manage privacy settings, and consent to data collection. |
| 9 | The app shall include educational resources about strep throat, covering symptoms, treatment options, and when to seek medical help. |
| 10 | This content will be accessible through a dedicated section in the app and integrated into the diagnostic feedback to provide context and guidance. |
| 11 | The app shall support user accounts, allowing users to securely log in, access their diagnostic history, and manage their personal information. |

6.1.2 Non-Functional Requirements

|  |  |
| --- | --- |
| 1 | The app shall have a clean, intuitive interface designed for ease of use, even for individuals without technical expertise. |
| 2 | The UI/UX design will prioritize clarity, ensuring that users can easily navigate the app, capture images, and understand diagnostic feedback. |
| 3 | The application shall perform efficiently, providing real-time diagnostic feedback with minimal delay. |
| 4 | The backend infrastructure will be scalable, capable of handling many concurrent users. |
| 5 | The mobile app will be developed using cross-platform tools such as React Native or Flutter, ensuring compatibility with both Android and iOS devices. |
| 6 | User data privacy will be a top priority, with clear policies and user consent mechanisms in place. |
| 7 | The app shall be compatible with both Android and iOS platforms, ensuring a wide reach and accessibility. |
| 8 | The design will account for differences in device capabilities, ensuring consistent performance across various smartphone models. |
| 9 | Educational Resources: Users can access a page of educational content. |
| 10 | Localization will extend to the educational content, instructions, and diagnostic feedback. |
| 11 | Diagnostic Feedback: The app will display diagnostic results directly on the user's device, offering clear and actionable feedback based on the analysis. |

6.2 System Architecture

The system architecture for the strep throat diagnostic tool comprises several key components, including the mobile application, backend server, database, and machine learning model. The architecture is designed to ensure efficiency, scalability, and security while providing a seamless user experience.

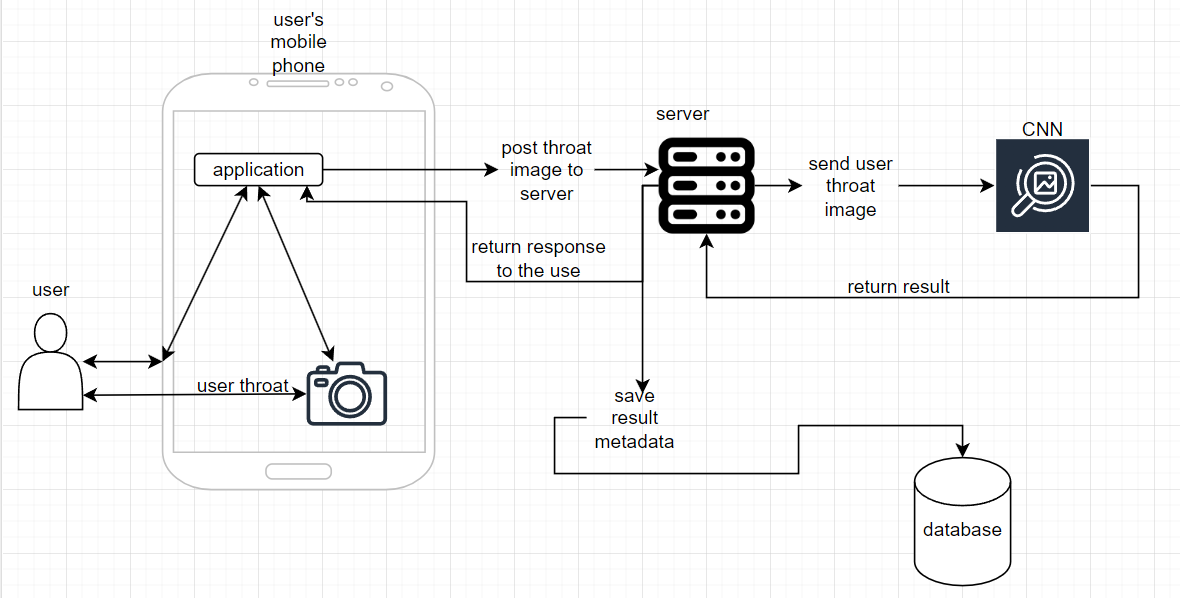


Figure 6: System Architecture

6.2.1 Mobile Application

The mobile application is the user-facing component of the system, designed to be intuitive and accessible. Key features include:

Image Capture Interface: The app will guide users through the process of capturing high-quality throat images, providing real-time feedback on image quality and positioning.

Diagnostic Feedback: The app will display diagnostic results directly on the user's device, offering clear and actionable feedback based on the analysis.

Educational Resources: Users can access a library of educational content, integrated with the diagnostic feedback to provide context and guidance.

The mobile app will be developed using cross-platform tools such as React Native or Flutter, ensuring compatibility with both Android and iOS devices.

6.2.2 Backend Server

The backend server handles the heavy lifting of image processing, model inference, and data management. Key responsibilities include:

Image Preprocessing and Analysis: The server receives images from the mobile app, preprocesses them to enhance quality, and runs them through the CNN model for diagnosis.

Model Inference: The trained CNN model is hosted on the backend, where it performs real-time analysis of the preprocessed images to generate diagnostic results.

Data Management: The backend manages user data, including images, diagnostic results, and account information. It ensures secure storage and retrieval of data, adhering to privacy regulations.

The backend will be developed using robust frameworks such as Node.js or Django, with cloud-based deployment to ensure scalability and reliability.

6.2.3 Database

The database is a critical component for storing and managing user data, including images, diagnostic results, and personal information. Key features include:

Secure Storage: The database will use encryption to protect sensitive data, with access controls in place to prevent unauthorized access.

Scalability: The database will be designed to handle a growing volume of data as the user base expands, with cloud-based solutions such as MongoDB or PostgreSQL being considered.

Data Retrieval and Analysis: The database will support efficient data retrieval for both user queries and model retraining purposes, enabling continuous improvement of the diagnostic tool.

6.2.4 Machine Learning Model

The CNN model is the core of the diagnostic tool, responsible for analyzing throat images and providing diagnostic feedback. Key considerations include:

Model Training: The CNN will be trained on a diverse dataset of labeled throat images, ensuring high accuracy and robustness in diagnosing strep throat.

Model Deployment: The trained model will be deployed on the backend server, optimized for real-time inference to provide immediate results to users.

Continuous Learning: The system will include mechanisms for updating and retraining the model with new data, improving accuracy over time.

The model will be developed using frameworks such as TensorFlow or PyTorch, with considerations for optimizing performance on mobile hardware.

6.3 User Interface and Experience

The success of the application heavily relies on its user interface (UI) and user experience (UX). The app is designed to be accessible to a broad audience, including those without technical expertise. Key design principles include:

Simplicity and Clarity: The interface will be straightforward, with clear instructions and minimal steps required for image capture and diagnosis.

Real-Time Guidance: Users will receive immediate feedback during image capture, ensuring that they take high-quality images suitable for analysis.

Educational Integration: The app will seamlessly integrate educational content with diagnostic feedback, helping users understand their results and next steps.

6.4 Testing and Quality Assurance

Quality assurance will be a critical part of the product development process. The app will undergo rigorous testing to ensure it meets all functional and non-functional requirements. Key testing phases include:

Unit Testing: Individual components of the app, such as image capture, preprocessing, and model inference, will be tested in isolation to ensure they function correctly.

Integration Testing: The app will be tested to ensure that all components work together seamlessly and that data flows correctly between the mobile app, backend server, and database.

User Acceptance Testing (UAT): The app will be tested by a group of end-users to ensure it meets their needs and expectations. Feedback from UAT will be used to make final adjustments before launch.

Performance Testing: The app will be tested under various conditions to ensure it performs well, even under high user loads.

6.5 Continuous Improvement and Updates

To ensure the app remains effective and up to date, the product will include a plan for continuous improvement. This will involve:

User Feedback: Collecting and analyzing user feedback to identify areas for improvement.

Model Updates: Periodically retraining the CNN model with new data to enhance diagnostic accuracy.

Feature Enhancements: Adding new features based on user needs and emerging healthcare trends, ensuring the app continues to provide value.

This comprehensive approach to product development ensures that the strep throat diagnostic tool will be robust, reliable, and capable of meeting the needs of both users and healthcare providers. By focusing on user experience, security, and continuous improvement, the product aims to make a meaningful impact on healthcare accessibility and outcomes.

6.6 Diagrams

* + 1. Use Case

תמונה שמכילה טקסט, תרשים, עיגול, צילום מסך

התיאור נוצר באופן אוטומטי

Figure 7: Use Case Diagram

* + 1. Sequence Diagram

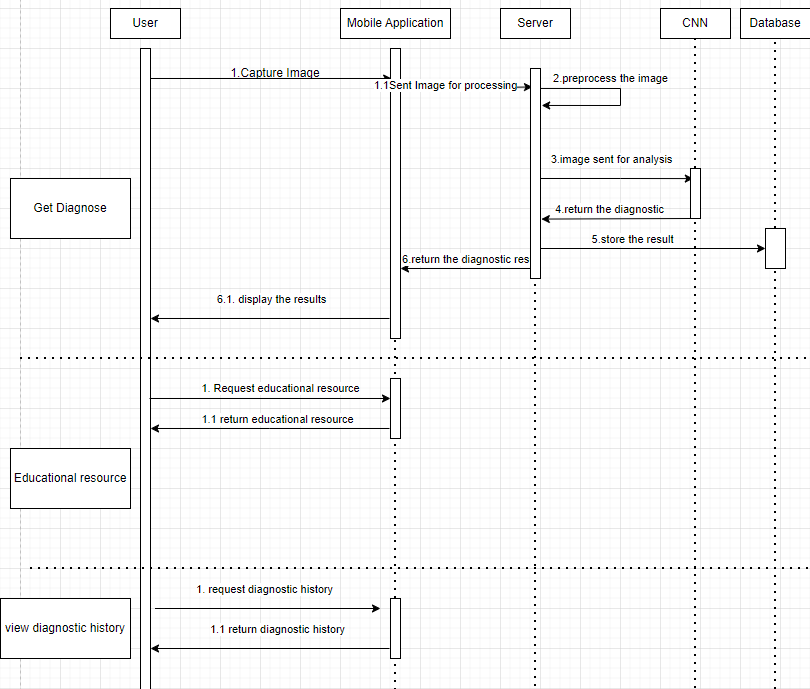


Figure 8: Sequence Diagram

* + 1. Activity Diagram

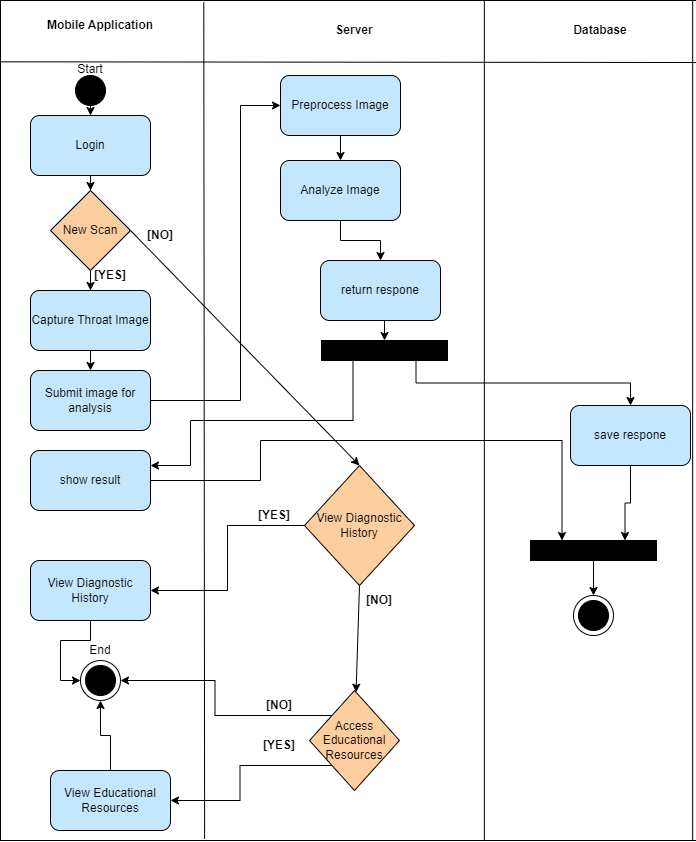
1. 

Figure 8: Activity Diagram

**7. Verification and Evaluation**

7.1 Evaluation

The primary goal of this project is to develop a reliable and accurate diagnostic tool for strep throat that can be used by both healthcare professionals and the public. The evaluation of this tool will focus on its ability to correctly diagnose strep throat by analyzing throat images captured using a smartphone. The following key metrics will be used to evaluate the performance of the application:

Diagnostic Accuracy:

The tool will be evaluated based on its ability to correctly classify images as either indicative of strep throat or not. The primary metric for this evaluation will be the accuracy, precision, recall, and F1-score of the machine learning model. These metrics will indicate how well the tool distinguishes between strep throat and other conditions.

The application will be tested on a separate validation dataset that was not used during the training phase to ensure unbiased evaluation. This dataset will include a variety of images representing different demographics, lighting conditions, and image qualities.

Speed and Efficiency:

The tool’s ability to provide real-time diagnostic feedback will be evaluated. The time taken from image capture to the delivery of diagnostic results should be minimal, ensuring that users receive timely feedback.

The performance of the backend infrastructure in handling concurrent user requests will also be evaluated to ensure that the application remains responsive under high loads.

User Experience and Accessibility:

User satisfaction and ease of use will be key evaluation criteria. The tool should be intuitive and easy to navigate, even for users without technical expertise. User feedback will be collected to assess the overall experience, including the clarity of instructions, ease of image capture, and understanding of diagnostic feedback.

The effectiveness of the educational content integrated within the app will also be evaluated, ensuring that users gain a clear understanding of strep throat and the importance of seeking medical advice if necessary.

Data Security and Compliance:

The application’s adherence to data protection regulations such as GDPR and HIPAA will be evaluated. This includes the secure handling, storage, and transmission of user data.

The effectiveness of data encryption, user authentication, and consent mechanisms will be tested to ensure the privacy and security of user information.

Model Reliability:

The reliability of the CNN model in maintaining accuracy over time will be evaluated by periodically retraining the model with new data. The goal is to ensure that the model remains effective as new variations of throat images are introduced.

7.2 Verification

Verification of the diagnostic tool will involve a comprehensive testing plan that ensures all components function correctly and meet the project’s requirements. The verification process will be divided into key areas: Mobile Application, Machine Learning Model, Backend Server, and Database. Each module will undergo rigorous testing, including unit tests, integration tests, and end-to-end tests.

7.2.1 Testing Plan

The iterative development process will include the following testing stages:

Mobile Application Testing:

Unit Tests: Individual components of the mobile app, such as the image capture interface, and real-time feedback, will be tested using frameworks like Jest and Mocha.

User Interface (UI) Testing: The UI will be tested for responsiveness, ease of navigation, and clarity of instructions. Manual QA testing will be conducted to assess user experience and ensure that the app is intuitive for all users.

Performance Testing: The app’s ability to handle high volumes of image processing requests will be tested. This includes evaluating the app’s performance on different devices and operating systems to ensure consistent behavior across platforms.

Machine Learning Model Testing:

Confusion Matrix Analysis: The CNN model will be evaluated using a confusion matrix, which will help assess the model’s accuracy, precision, recall, and F1-score in classifying throat images.

Cross-Validation: The model will undergo cross-validation to ensure it generalizes well to new data. This involves training the model on different subsets of the dataset and validating it on the remaining data.

Model Stress Testing: The model will be tested under various scenarios, including different image qualities, lighting conditions, and throat conditions, to evaluate its robustness.

Backend Server Testing:

API Testing: The backend APIs responsible for image processing, model inference, and data management will be tested for correctness, efficiency, and security. Tools like Postman will be used to simulate API requests and evaluate responses.

Load Testing: The server’s ability to handle concurrent requests will be tested to ensure scalability. Load testing tools like JMeter will be used to simulate high user traffic and evaluate server performance.

Data Management Testing: The correctness of data storage, retrieval, and management will be verified, ensuring that user data is accurately handled and securely stored.

Database Testing:

Data Integrity Testing: The database will be tested to ensure that all user data, including images and diagnostic results, are stored and retrieved accurately. This includes verifying data consistency across different scenarios.

Scalability Testing: The database’s ability to scale with growing user data will be tested. This includes evaluating the performance of database queries under increased load.

7.2.2 Test Cases

Specific test cases will be designed to verify each module’s functionality. Examples include:

|  |  |  |  |
| --- | --- | --- | --- |
| Test # | Module | Tested Function | Steps (What to press) |
| 1 | Mobile Application | Image Capture Interface | Open app, tap "Capture Image" button |
| 2 | Mobile Application | Diagnostic Feedback | After capturing an image, tap "Analyze" button |
| 3 | Mobile Application | Image Quality Feedback | Tap "Capture Image" with poor lighting |
| 4 | Mobile Application | Account Creation | Tap "Sign Up," fill out details, and press "Submit" |
| 5 | Mobile Application | Login and Logout Functionality | Tap "Log In," enter credentials, and press "Login" |
| 6 | Mobile Application | Educational Resources Navigation | Tap "Educational Resources" tab |
| 7 | Machine Learning Model | Confusion Matrix Analysis | Run diagnostic on test images |
| 8 | Machine Learning Model | Model Generalization | Test with images under varied lighting conditions |
| 9 | Backend Server | API Response Time | Tap "Analyze" after image capture |
| 10 | Backend Server | Load Handling | Simulate 100 concurrent image analysis requests |
| 11 | Database | Data Integrity | Check stored results by accessing "History" tab |
| 12 | Database | Encryption | Inspect data transmission when analyzing an image |
| 13 | Database | Scalability | Simulate storage of a large number of user records |

This detailed testing and verification process will ensure that the diagnostic tool is robust, reliable, and ready for real-world use. By thoroughly evaluating both the functionality and performance of each component, the project aims to deliver a high-quality product that meets user needs and exceeds industry standards.

1. **References**:
2. John V. Ashurst; Laura Edgerley-Gibb., "Streptococcal Pharyngitis" (2023).

<https://www.ncbi.nlm.nih.gov/books/NBK525997/>

1. Robert Luo., "Diagnosis and Management of Group a Streptococcal Pharyngitis in the United States, 2011–2015" (2019).

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6390592/>

1. Laura Norton et al., "The treatment of streptococcal tonsillitis/pharyngitis in young children" (2021).

<https://www.sciencedirect.com/science/article/pii/S2095881121000524>

1. Centers for Disease Control and Prevention., "About Strep Throat" (2024).

<https://www.cdc.gov/group-a-strep/about/strep-throat.html>

1. Tae Keun Yoo et al., "Toward automated severe pharyngitis detection with smartphone camera using deep learning networks" (2020).

<https://www.sciencedirect.com/science/article/pii/S0010482520303115>

1. Askarian, B., Yoo, S.-C., Chong, J.W., "Novel Image Processing Method for Detecting Strep Throat Using Smartphone" (2019).

<https://pubmed.ncbi.nlm.nih.gov/31357633/>

1. Chin, S.Y., et al., "Bacterial Image Analysis Using Multi-Task Deep Learning Approaches for Clinical Microscopy" (2024).

<https://peerj.com/articles/cs-2180/>

1. Zahid Mustafa et al., "Diagnostic Methods, Clinical Guidelines, and Antibiotic Treatment for Group A Streptococcal Pharyngitis: A Narrative Review" (2020).

<https://www.frontiersin.org/journals/cellular-and-infection-microbiology/articles/10.3389/fcimb.2020.563627/full>

1. Priyanka Kumari., "Skin Cancer Detection Model Using CNN: A Comprehensive Guide" (2024).

<https://www.labellerr.com/blog/detecting-skin-cancer-using-convolutional-neural-networks-a-comprehensive-guide/>

1. Dimpy Varshn., "Pneumonia Detection Using CNN based Feature Extraction" (2020).

<https://ieeexplore.ieee.org/document/8869364>

1. Harry Pratt., "Convolutional Neural Networks for Diabetic Retinopathy" (2016).

<https://www.sciencedirect.com/science/article/pii/S1877050916311929>