Progress report submitted to
the Department of Computer Engineering of Çankaya University
in partial fulfillment of the requirement for
CENG 407 Innovative System Design and Development I

Gluten Detection System

By;

Alara KANTÜRK - 201812031

Ecenaz DOĞAN - 202011060

Ege MEZGİT - 201812034

Elif Bengü SARAÇ - 201911053

Yıldız Pelinsu KÖSEDAĞI - 201911037

Zeynep SERTEL – 201811052

Group No: 07

Supervisor: Dr. Instructor Serdar ARSLAN

Çankaya University

Department of Computer Engineering

2023 - 2024

Table of Contents

List of Figures

Introduction

The Gluten Detection System stands as an innovative solution designed to address the persistent challenges faced by individuals afflicted with gluten-related disorders, notably Celiac Disease. This intricate and chronic sensitivity to gluten, predominantly found in grains like Wheat, Barley, Rye, and Oats, necessitates stringent adherence to a gluten-free diet, making the identification of gluten in food products of paramount importance.

This comprehensive system leverages advanced deep learning algorithms and machine learning models to offer a streamlined method for users to detect gluten content in various food items. The mobile application interface, compatible with both Android and IOS platforms, allows seamless scanning of product barcodes or ingredients, providing instantaneous feedback regarding the presence of gluten or allergens.

The project's foundation lies in a documented Software Requirements Specification (SRS) and Software Design Document (SDD), outlining the system's intricacies, functional boundaries, and technological interfaces. These documents serve as guiding principles for developers and stakeholders, ensuring comprehensive understanding and comprehensive insight throughout the project.

Beyond its immediate impact on individuals managing gluten-related ailments, the Gluten Detection System bears broader implications for healthcare professionals, nutritionists, and the wider food industry. By fostering a safer and more inclusive food environment, this system sets a new standard for food safety technology.

The meticulous design, rigorous technological framework, and commitment to enhancing user accessibility through intuitive interfaces underscore the Gluten Detection System's significance in revolutionizing dietary management for individuals with gluten sensitivities.

A Review of the Computer Science Literature Relating to Gluten Detection System

Abstract

Celiac disease: It occurs with sensitivity to a protein called gluten, found in foods such as wheat, barley, and rye [1]. It is an immune system disease and can occur at any age [1]. There is no definitive treatment for celiac disease and people must follow a gluten-free diet throughout their lives. According to the Ministry of Health data, there are 154,027 celiac patients in Turkey as of the end of 2022 [2]. People with this disease must live by checking whether the foods they consume contain gluten. With the Gluten Detection System, individuals with celiac disease and people who want to follow a gluten-free diet can access gluten-free products. Our aim here is to develop an application that will help users, such as celiac patients, who want to stay away from certain ingredients in foods that they should not consume or do not prefer.

Öz

Çölyak hastalığı; besinlerdeki buğday, arpa ve çavdarda bulunan gluten adlı bir proteine karşı hassasiyet ile ortaya çıkar [1]. Bir bağışıklık sistemi hastalığıdır ve her yaşta görülebilir [1]. Çölyak hastalığının kesin bir tedavisi yoktur ve kişiler ömürleri boyunca glutensiz beslenmek zorundadır. Sağlık bakanlığı verilerine göre 2022 yılı sonu itibari ile Türkiye'de 154.027 çölyak hastası bulunmaktadır [2]. Bu hastalığa sahip kişiler tükettikleri besinlerin gluten içerip içermediğine bakarak yaşamalıdır. Gluten Algılama Sistemi ile çölyak hastası olan bireyler ve glutensiz beslenmek isteyen kişiler, glutensiz ürünlere ulaşabilir. Buradan hareketle amacımız çölyak hastaları gibi besinlerdeki belirli içeriklerden tüketmemesi gereken yada tercih etmedikleri içeriklerden uzak durmak isteyen kullanıcılara kullanımı kolay bir uygulama geliştirmektir.

1. Introduction

1.1. Problem Statement

Celiac Disease is a lifelong and chronic allergy and sensitivity of the small intestine to the protein called "gluten". Grains such as Wheat, Barley, Rye and Oats contain gluten. When Celiac Patients consume gluten-containing foods, nutritional deficiency due to allergy occurs in the intestinal mucosa, followed by disease symptoms. The only treatment for celiac disease is a strict gluten-free diet. With strict implementation of the diet, the flattened small intestine surface regains its normal shape and function. Too little amount of gluten causes repeated damage to the intestines. Disruption or abandonment of the diet can cause serious diseases that are much more difficult to treat. These diseases cannot be treated with bio resonance or alternative medicine methods. The only known and proven treatment method today is the gluten-free diet.[3] As we mentioned, it is vital for celiac patients to follow their diet correctly and to consume packaged products by examining their contents. We will implement the Gluten Detection System to eliminate their grievances regarding food, to end the waste of time and effort spent reading content for hours in the markets, and to make their lives more practical.

1.2. Solution Statement

In this project, a Gluten Detection System will be developed to find information about the diseases of gluten-free products. Our practice can also be transmitted to other allergens. This application aims to make the user’s experience with grocery products more practical and effective. The mobile application allows users to easily search if that product contains any allergens. As a result of this, the application will inform the user about prohibited content, if any. Additionally, our artificial intelligence aims to improve user experience by showing users the products they can access. This helps the user not miss gluten or any allergens they initially selected. In summary, Gluten Detection System will help celiac patients.

1.3. Project Plan

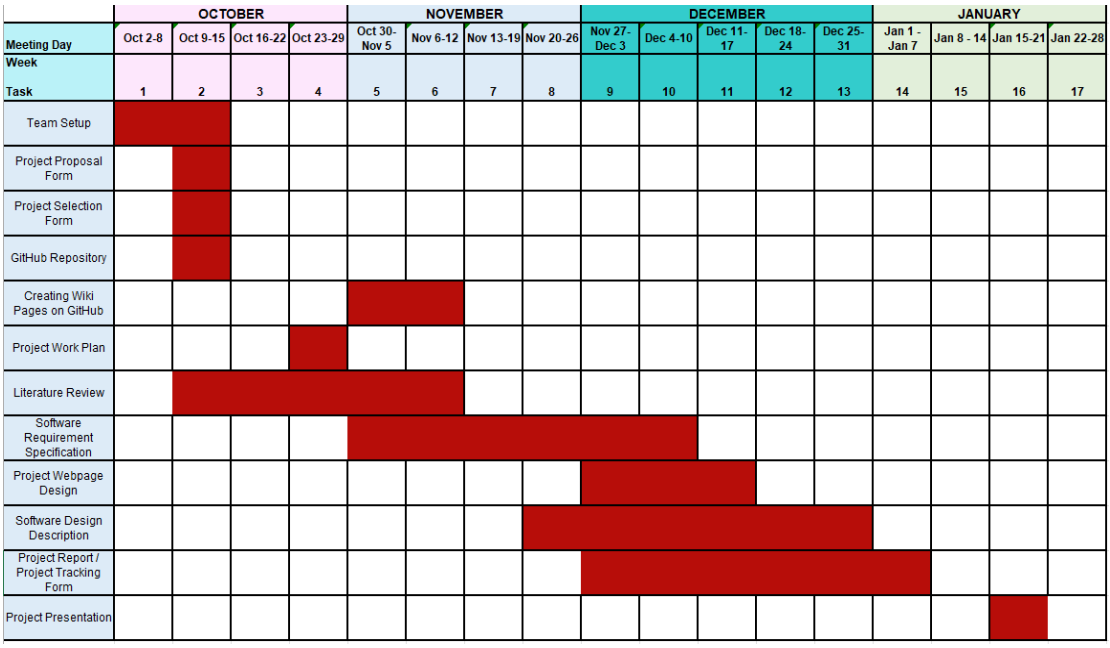

Figure 1: Project Plan

2. Machine Learning

2.1. What is Machine Learning?

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy [4]. Machine Learning is a concept that allows the machine to learn from examples and experience, and that too without being explicitly programmed [5]. Machine learning can be broadly categorized into the following types:

- Supervised Learning
- Unsupervised Learning
- Semi-supervised Learning
- Reinforcement Learning

2.1.1. Supervised Learning

The machine learns from labeled data. Normally, the data is labeled by humans [6]. The objective is to train the algorithm to predict accurate labels for new, unseen data [7]. Examples of supervised learning algorithms include:

- Decision Trees
- Support Vector Machines
- Random Forests

2.1.2. Unsupervised Learning

In this machine learning approach, algorithms analyze unlabeled data without predefined output labels. The objective is to discover patterns, relationships, or structures within the data [7]. Examples of unsupervised learning algorithms include:

- K-means
- Hierarchical clustering

2.1.3. Semi-supervised Learning

Semi-supervised learning is a hybrid machine learning approach that combines labeled and unlabeled data for training [7].

2.1.4. Reinforcement Learning

The machine learns through a reward-based system [6].

2.2. Machine Learning Algorithms

Linear regression: This algorithm is used to predict numerical values, based on a linear relationship between different values [4].

Logistic regression: This supervised learning algorithm makes predictions for categorical response variables, such as “yes/no” answers to questions [4].

Decision trees: Decision trees can be used for both predicting numerical values (regression) and classifying data into categories. Decision trees use a branching sequence of linked decisions that can be represented with a tree diagram [4].

Random forests: In a random forest, the machine learning algorithm predicts a value or category by combining the results from several decision trees [4].

K-Means: It is an unsupervised learning algorithm that solves clustering problems [4]. It aims to group data points based on their proximity to one another [8].

Support vector machine (SVM): A support vector machine (SVM) is a supervised learning algorithm commonly used for classification and predictive modeling tasks [8].

3. Deep Learning

3.1. What is Deep Learning?

Deep Learning is a subfield of Machine Learning that involves the use of neural networks to model and solve complex problems [9]. Neural networks attempt to model human learning by digesting and analyzing massive amounts of information, also known as training data [10]. Deep learning models can recognize complex patterns in pictures, text, sounds, and other data to produce accurate insights and predictions [11].

3.2. Deep Learning Models

Deep learning models are based on artificial neural networks (ANN). In deep learning models, multi-layer artificial neural networks (MLP) and complex mathematical operations are used. Deep learning models include predefined sets of steps (algorithms) that tell the file how to treat certain data [10]. Deep learning algorithms can automatically learn and improve from data without the need for manual feature engineering [9]. Deep learning models are suitable for recognizing more complex patterns in text, images, or sounds.

Some of deep learning models are listed below:

- Artificial Neural Networks (ANN)
- Convolutional Neural Networks (CNN)
- Recurrent Neural Networks (RNN)
- Long Short-Term Memory Networks (LSTM)

- Transformers
- Gated Recurrent Unit Networks (GRU)

3.2.1. Artificial Neural Networks (ANN)

Artificial Neural Networks (ANN) are multi-layer fully connected neural networks. ANNs are very flexible yet powerful deep learning models. They can model any complex function [12].

3.2.2. Convolutional Neural Networks (CNN)

A CNN is a multilayer neural network that was biologically inspired by the animal visual cortex [13]. It is used in areas such as image processing, image analysis and object detection.

3.2.3. Recurrent Neural Networks (RNN)

Recurrent Neural Networks (RNNs) predict the next point in a sequence by using a series of structures one after the other [14]. RNNs are mostly used in captioning the image, time series analysis, recognizing handwritten data, and translating data to machines [15].

3.2.4. Long Short-Term Memory Networks (LSTM)

Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) that is specifically designed to handle sequential data, such as time series, speech, and text [16]. LSTM networks are capable of learning long-term dependencies in sequential data, which makes them well suited for tasks such as language translation, speech recognition, and time series forecasting [16].

3.2.5. Transformers

A transformer model is a type of deep learning model. It is especially used in the field of Natural Language Process (NLP). The model was first described in a 2017 paper called "Attention is All You Need" by Ashish Vaswani, a team at Google Brain, and a group from the University of Toronto [17]. The transformer is a neural network that can learn the context in sequential data such as languages [18]. Transformers were developed to solve the problem of sequence transduction, or neural machine translation. That means any task that transforms an input

sequence to an output sequence. This includes speech recognition, text-to-speech transformation, etc. [19].

OpenAI's popular ChatGPT text generation tool makes use of transformer architectures for prediction, summarization, question answering and more, because they allow the model to focus on the most relevant segments of input text. The BERT model, or Bidirectional Encoder Representations from Transformers, is based on the transformer architecture [17].

3.2.6. Gated Recurrent Unit Networks (GRU)

GRU networks are a type of RNN. GRU has a similar structure to LSTM. Like LSTM, GRU can process sequential data such as text, speech, and time-series data [20].

3.3. Applications of Deep Learning

Computer vision: Computer vision is the computer's ability to extract information and insights from images and videos [11]. Deep learning models can enable machines to identify and understand visual data [9]. Computer vision has several applications, such as the following:

- Object detection and recognition
- Image classification

Recommendation engines: Applications can use deep learning methods to track user activity and develop personalized recommendations [11].

Natural language processing (NLP): Deep learning model can enable machines to understand and generate human language [9]. Some of the main applications of deep learning in NLP include:

- Language translation
- Sentiment analysis
- Speech recognition

3.4. Difference between Machine Learning and Deep Learning

- In machine learning, it is possible to work with smaller data sets. In deep learning, data sets are larger than in machine learning.

- Deep learning models tend to perform well with a large amount of data, while old machine learning models may stop improving after reaching a saturation point [21].
- Feature extraction is done by humans in machine learning whereas deep learning models figure it out by themselves [21].

3.5. Deep learning for OCR

Optical Character Recognition (OCR) is a technology that aids users to recognize characters using optical mechanisms. It works as the reading ability of a human, but it cannot be compared with the human capability. This technology can recognize printed and handwritten text together. If the quality of the input text is not good enough to ease recognition, the performance of OCR technology decreases. Tesseract, Google's open-source OCR engine, language translation service, and text-to-speech synthesizer for Android operating system help users to use their phones to read text loudly, using input image with camera or digital scanner.[22]
There are some steps to recognize a text. These are [22]:

- **Scanning** is needed to change the original document to its digital image, using an Android mobile camera. Using sensing devices, OCR scanner changes the light intensity into grayish levels and turns the document into bilevel image of white and black [23].
- **Segmentation** is needed to locate regions of handwritten or printed text. It divides characters or words.
- **Preprocessing** is needed to smooth and normalize the digital image, using filling and thinning operations.
- **Feature Extraction** is needed to extract the symbols' features, characteristics.
- **Recognition** is needed to understand the characters word by word, using lines.

Deep Learning algorithms can be seen for OCR processes. To understand the text, Convolution Neural Network (CNN), Connectionist Temporal Classification (CTC), Recurrent Neural Network (RNN), Long Short-Term Memory Networks (LSTM) deep learning constructions can be used to overcome with segmentation, extraction and handwritten features. To classify the characters, the image is selected with the desired height and width.
After reaching the required height and width, the image passes through the convolutional layers that include the different number of filters to control the required features in CNN [24]. Also, OCR engines may remember the previous text and guess the coming data with RNN [25]. With these methods, text recognition can be easier to use.

There might be provided data as training data to compare and recognize the input characters. Including all letters and characters into a text with their different fonts, credit cards with different face patterns, license plates can be used as a dataset [26]. If the character or symbol cannot be recognized, OCR engine adds them into data pool as a new dataset.

4. Digital Image Process

4.1. What is Digital Image Process?

Image processing on digital images using computer algorithms is called digital image processing. A sub-subject of image processing is digital signal processing and a sub-subject of digital image processing. So, image processing is a more general approach than numerical processing. Digital image processing processes numerical data and performs mathematical operations. It is used in many industrial applications such as medicine, military industry, transportation and transportation robotics, biomedical fields, oil exploration, environmental monitoring, and mapping.[27]

4.2. How Digital Image Processing is Implemented?

A digital image is an image that consists of numerical values and can be displayed in a computer environment. The image is assigned to a function through algorithms and processing is done through this function. An image can be defined as a two-dimensional function f(x, y). (x, y) is called the intensity or gray level of the image at that point.[28] With image processing techniques, improved or different images can be obtained by using digital image data and object recognition processes can be performed.[29] Image processing steps consist of different techniques depending on the intended use, but it starts with the capture of the image.

4.3. Digital Image Processing Techniques

Image Filtering: Filters are used on images to reduce noise or remove certain features.

Feature Extraction: The process of recognizing and extracting important features from images.

Object Recognition and Classification: The process of recognizing objects in images and assigning them to specific classes.

Segmentation: The process of dividing an image into different sections or objects.

Pattern Matching: Finding a specific pattern or object in an image.

Calculation Operations: Performing mathematical operations on images such as pixel operations, contrast adjustment.

Edge Detection: Used to determine the boundaries of objects in the image.

Motion Detection: The process of detecting and tracking the movement of objects in the image.

Face Recognition: It is the process of recognizing and classifying the human face.

4.4. Digital Image Processing and Artificial Intelligence

Machine learning and deep learning offer many benefits and application areas in digital image processing. Artificial intelligence is used in image processing examples such as image segmentation, object detection, pattern recognition, anomaly detection. Some algorithms used in this field are as follows:

GMM (Gaussian Mixture Model): It is a machine learning algorithm used to divide pixels into parts. It is used for image segmentation.

Optical Flow: Used to track the movement of objects.

YOLO (You Only Look Once): Used for object recognition and classification.

K-Means Clustering: Used to group similar pixels.

Convolutional Neural Networks (CNN): This model is used to perform image, face, object recognition and segmentation. This model is a powerful algorithm for visual processing.

Generative Adversarial Networks (GAN): Used for image generation and correction.

5. Barcode Detection using Deep Learning Techniques

This report examines gluten scanning applications, which aim to quickly and accurately identify gluten-containing products and help individuals with gluten intolerance or celiac disease, consume food safely. It also provides recommendations for the development of these applications by comparing libraries used on different platforms.

5.1 Gluten Scanner Applications and Use Cases

Gluten scanning applications allow users, especially those with gluten intolerance or celiac disease, to safely review food products before consumption. Through these applications, users

can instantly obtain information about the content of products by scanning the barcodes on them. Therefore, gluten scanning applications have a range of different use cases, such as:

- Evaluating meal options in restaurants
- Checking the content of products while grocery shopping
- Gluten Scanner Libraries and Comparisons

In this section, three fundamental libraries that can be used when developing gluten scanning applications are compared:

- **ZXing (Zebra Crossing):** It can be used in different languages like Java, Python, and JavaScript. It supports various barcode types such as QR codes, EAN-13, and UPC. It can be used on different platforms, including Android, iOS, and the web. It is an open-source library with a wide user base.
- **ZBar:** It is compatible with languages like C and Python. It recognizes common barcode types such as QR codes, EAN-13, and UPC. It can be easily integrated into different platforms. It is lightweight, performance-oriented, and open-source.
- **QuaggaJS:** It is compatible with JavaScript and is ideal for web applications. It recognizes barcode types like QR codes, EAN-13, and UPC. It provides a fast and user-friendly solution.

Comparison of Gluten Scanner Applications:

- **ZXing vs. ZBar:** Both libraries have a broad user base and can be used on various platforms. The choice between these two libraries depends on the needs of your project and platform requirements.
- **QuaggaJS:** If you are developing a web-based application, QuaggaJS offers a user-friendly and fast solution.

5.2 Deep Learning Frameworks

In addition to these libraries, we should also mention deep learning frameworks. Deep learning frameworks like TensorFlow, PyTorch, or Keras are used by the majority of barcode detecting applications. These frameworks offer a large selection of pre-trained models as well as resources for creating unique deep learning models.

- **TensorFlow**

Definition: Google's open-source TensorFlow machine learning framework offers great flexibility in the creation and training of deep learning models. It offers support for scalability across multiple platforms and has a sizable user base. TensorBoard is included for performance assessment.

Application to Barcode Detection: TensorFlow is used to create personalized CNN-based models that find and identify barcodes in pictures or video frames. Its adaptability enables customization for certain use cases.

- **PyTorch**

Definition: The open-source PyTorch framework is a product of Facebook's AI Research Lab (FAIR) and is renowned for its easily navigable debugging facilities and dynamic computation graph. In academic and scientific environments, it is becoming more and more popular, although possibly having a smaller community.

Use in Barcode Detection: PyTorch's dynamic computing graph makes it a good choice for research- and experiment-focused programming when it comes to barcode detection. It offers customization options for models.

- **Keras**

Definition: The open-source Keras deep learning framework aims to be user-friendly. It can be used with various backends, such as TensorFlow. It makes model creation, training, and evaluation easier by providing a high-level API.

Use in Barcode Detection: If you're looking for a user-friendly interface to create barcode detection models, Keras is a great option. Due to its vast capability, TensorFlow is frequently used as its backend.

6. Gluten Detection System

The Gluten Detection System is an application that is designed to simplify the process of identifying gluten in products. This application leverages the power of technology to provide users with an easy and accurate solution for gluten detection. By simply scanning a product’s barcode or the ingredients using a smartphone or a tablet, the application analyses the ingredients using deep learning and/or machine learning models and promptly alerts the user of

the presence of gluten. This real-time feedback empowers individuals with gluten-related illnesses or dietary concerns such as people with celiac disease or gluten sensitivity and helps them make informed choices while shopping or dining in their daily lives.

The Gluten Detection System differs from other applications in this area by using deep learning algorithms and models so in the case of a deficiency in the dataset for a certain product, the application can determine whether the product is safe for consumption or not. The users no longer need to rely solely on manual label or ingredient reading that can be time consuming and error prone. The Gluten Detection System offers a convenient and reliable way to enhance food safety and dietary preferences, giving users a peace of mind, they deserve in their quest for a gluten-free life.

This application is not only a life changer for individuals with gluten related problems but also for healthcare professionals, nutritionists or the food industry as a whole, contributing to a safer and more inclusive food environment. The Gluten Detection System represents a step forward in the search for a gluten-free life, delivering the peace of mind and confidence the users deserve when it comes to their dietary choices.

6.1. Related Works

Numerous applications akin to our Gluten Detection Application have been developed, such as "The Celiac App," "Glutensiz Nokta," "Gluten Tarayıcı," "Gluten Free Scanner," "Find Me Gluten Free," "Is It Gluten Free?", "Celiac UK" and more. However, our research has revealed that most of these applications, while serving a great purpose, display certain limitations and problems. A prevalent issue we have identified is the lack of comprehensive support for the Turkish language and products, making them more suitable for international usage rather than local usage such as Gluten Free Scanner having a huge up-to-date growing database with 500.000+ products of food and drink of the US [31] but not having the products of Turkey nor having Turkish language support. Furthermore, among the available applications, only “Glutensiz Nokta” seems tailored for local use. Regrettably, upon closer examination, this application turned out to be lacking in some respects. It suffers from a limited product database and its accuracy in determining gluten content has been found wanting at times, providing users with inaccurate information.

Our Gluten Detection Application seeks to address these observable gaps in existing solutions by offering a variety of algorithms and models in deep and machine learning so in the case of a deficiency in the dataset, the system will still work by analyzing the product on its own and

will provide an unmatched commitment to precision in gluten detection. By providing comprehensive support for the Turkish language and an extensive array of products, we aim to bridge the gap between international and local usage, ensuring that our application serves the specific needs of the Turkish-speaking community. Our strict research and development process aims to gather a wealth of data, enabling us to deliver a superior and more accurate tool for those who demand reliable gluten detection in their dietary choices. Through these efforts, we aim to set a new standard for localized gluten detection applications, thus enhancing the experience of users seeking to embrace a gluten-free lifestyle while making well-informed choices with utmost confidence.

7. Conclusion

In conclusion, this analysis of the literature has given a thorough understanding of the Gluten Detection System, an innovative application created to make it easier to identify gluten in food products. We have discovered both the advantages and disadvantages of the instruments available today for gluten detection through a review of the literature and applications that have already been developed. Our review emphasizes how important it is to have accessible and reliable gluten testing tools, especially for people who have celiac disease or gluten sensitivity. It is clear that many of the current programs, albeit useful, have some drawbacks, such as inaccurate gluten content analysis and restricted language compatibility. As this literature review discusses, the Gluten Detection Application shows promise in addressing these shortcomings with its deep learning/ machine learning models and full support for the Turkish language. By bridging the gap between local and worldwide usage, this application makes sure to meet the unique requirements of the Turkish-speaking community.

In summary, the Gluten Detection System, as explored in this literature review, offers a promising pathway toward a more inclusive, accurate, and accessible tool for gluten detection, with the potential to greatly benefit both individuals and the food industry as a whole. This review underscores the significance of this innovative application in the context of gluten-related dietary concerns and sets the stage for further research and development in this important area.

8. References

[1] “Çölyak Hastalığı Çeşitleri, Belirtileri ve Tedavisi,” acibadem.com.tr. https://www.acibadem.com.tr/ilgi-alani/colyak-hastaligi/. (accessed Oct. 30, 2023).

[2]“Metabolizma ve Çölyak,” hsgm.saglik.gov.tr. https://hsgm.saglik.gov.tr/tr/metabolizmave-colyak. (accessed Oct. 30, 2023).

[3] https://colyak.org.tr/colyak-nedir/

[4] “What is machine learning?,” ibm.com. https://www.ibm.com/topics/machine-learning (accessed Nov. 4, 2023).

[5] “Machine Learning Tutorial — Understand With Examples,” medium.com. https://medium.com/edureka/machine-learning-tutorial-f2883412fba1 (accessed Nov. 1, 2023).

[6] “A Beginner’s Guide to Machine Learning,” medium.com. https://medium.com/@randylaosat/a-beginners-guide-to-machine-learning-dfadc19f6caf (accessed Nov. 4, 2023).

[7] “Top 10 Machine Learning Algorithms For Beginners: Supervised, and More,” simplilearn.com. https://www.simplilearn.com/10-algorithms-machine-learning-engineers-need-to-know-article (accessed Nov. 4, 2023).

[8] “10 Machine Learning Algorithms to Know in 2023,” coursera.org. https://www.coursera.org/articles/machine-learning-algorithms (accessed Nov. 4, 2023).

[9] “Introduction to Deep Learning,” geeksforgeeks.org. https://www.geeksforgeeks.org/introduction-deep-learning/ (accessed Nov. 1, 2023).

[10] “What Is Deep Learning? Definition, Examples, and Careers,” coursera.org. https://www.coursera.org/articles/what-is-deep-learning (accessed Nov. 1, 2023).

[11] “What is Deep Learning? - Deep Learning Explained,” aws.amazon.com. https://aws.amazon.com/tr/what-is/deep-learning/ (accessed Nov. 1, 2023).

[12] “Applied Deep Learning - Part 1: Artificial Neural Networks,” towardsdatascience.com. https://towardsdatascience.com/applied-deep-learning-part-1-artificial-neural-networks-d7834f67a4f6 (accessed Nov. 4, 2023).

[13] “Deep learning architectures,” developer.ibm.com. https://developer.ibm.com/articles/cc-machine-learning-deep-learning-architectures/ (accessed Nov. 1, 2023).

[14] Ferdi Doğan and İbrahim Türkoğlu, “Derin Öğrenme Modelleri ve Uygulama Alanlarına İlişkin Bir Derleme,” DÜMF Mühendislik Dergisi, vol. 10, no. 2, pp. 409-445, 2019. Accessed: November 3, 2023. [Online]. Available: https://dergipark.org.tr/tr/download/article-file/738321

[15] “Deep Learning Algorithms,” javatpoint.com. https://www.javatpoint.com/deep-learning-algorithms (accessed Nov. 4, 2023).

[16] “Deep Learning | Introduction to Long Short Term Memory,” geeksforgeeks.org. https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/ (accessed Nov. 4, 2023).

[17] “What is a transformer model | IBM,” ibm.com. https://www.ibm.com/topics/transformer-model (accessed Nov. 4, 2023).

[18] “Introduction to Transformer,” medium.com. https://medium.com/intro-to-artificial-intelligence/introduction-to-transformer-cd511b7bf67f (accessed Nov. 4, 2023).

[19] “How Transformers Work,” towardsdatascience.com. https://towardsdatascience.com/transformers-141e32e69591 (accessed Nov. 4, 2023).

[20] “Gated Recurrent Unit Networks,” geeksforgeeks.org. https://www.geeksforgeeks.org/gated-recurrent-unit-networks/ (accessed Nov. 4, 2023).

[21] “Deep learning series 1: Intro to deep learning,” medium.com. https://medium.com/intro-to-artificial-intelligence/deep-learning-series-1-intro-to-deep-learning-abb1780ee20 (accessed Nov. 3, 2023).

[22] Optical Character Recognition, Ravina Mithe, Supriya Indalkar, Nilam Divekar, (International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-2, Issue-1, March 2013).

[23] Optical Character Recognition, Line Eikvil, December 1993.

[24] Telugu Optical Character Recognition Using Deep Learning, Gade Suresh, Ch.Rajendra Prasad, Dr. Sreedhar Kollem (2022 3rd International Conference for Emerging Technology (INCET) Belgaum, India. May 27-29, 2022).

[25] Quranic Optical Text Recognition Using Deep Learning Models, Masnizah Mohd, Faizan Qamar, Universiti Kebangsaan Malaysia under Grant DCP-2017-007/4, March 4, 2021.

[26] Improving Deep Learning based Optical Character Recognition via Neural Architecture Search, Zhenyao Zhao, Min Jiang∗, Senior Member, IEEE, Shihui Guo∗, Member, IEEE Zhenzhong Wang, Fei Chao and Kay Chen Tan, Fellow, (IEEE Authorized licensed use limited to: ULAKBIM UASL - Cankaya University. Downloaded on November 01,2023 at 19:04:15 UTC from IEEE Xplore).

[27] M.Karakoç. (2012,February). “GÖRÜNTÜ İŞLEME,TEKNOLOJİLER VE UYGULAMALARI” Available: https://ab.org.tr/ab12/sunum/21-goruntu_isleme-Karakoc.pdf [Accessed 2 November 2023].

[28] "Görüntü İşleme Nedir?," [Online]. Available: https://www.simurgai.com/G%C3%B6r%C3%BCnt%C3%BC-%C4%B0%C5%9Fleme-Nedir/ [Accessed 1 November 2023].

[29] G. Samtaş., M. Gülesin. (2011, May). "SAYISAL GÖRÜNTÜ İŞLEME VE FARKLI ALANLARDAKİ UYGULAMALARI". Electronic Journal of Vocational Colleges [Online]. vol. 2,issue 8. Available: https://dergipark.org.tr/tr/download/issue-full-file/5394

[30] Github. "zxing-android-embedded."Available: https://github.com/journeyapps/zxing-android-embedded.

[31] The Gluten Free Scanner [Online]. Available: https://www.scanglutenfree.com/. [Accessed 1 November 2023].

Software Requirements Specification (SRS)

1. Introduction

1.1 Purpose of this document

The purpose of this document is to define and explain the concepts and system requirements. This document aims to make it easier for the people in the development of this project and make it clear to someone who may be joining the project. It also makes sure to document every step of the project so any change that may be made is easier to follow. An SRS is a crucial part of software development.

1.2 Scope of your project

The scope of the project is to clearly define and provide a complete understanding of each boundary of the project. It outlines basic properties to our project such as objectives, deliverables, functional and nonfunctional requirements, inclusions and exclusions, constraints, assumptions, dependencies, boundaries, and project planning.

2. General Description

2.1 Glossary

- SRS: Software Requirements Specification
- IOS: iPhone Operating System
- UI: User Interface
- CLI: Command Line Interface
- IPC: Inter Process Communication
- API: Application Programming Interface
- QR Code: Quick Response Code
- DFD: Data Flow Diagram

2.2 User Characteristics

The system has one component that is User.
USER: The users can scan the products of their choice, make an account, login, edit their accounts and delete their accounts.

2.3 Overview of Functional Requirements

User registration and sign in: The system allows users to create an account by entering their username, e-mail address and password.
Users: They can scan the product's package and get content information about the product, listen to this content information in an audible way, and add the product to the favorites list.They can edit the information in their profile.
System: It contains a database where data is stored

2.4 General Constraints and Assumptions

- Image limitation, image defects due to the label of the product.
- Languages Differences, the browser will only detect Turkish.
- The applied algorithm gives inaccurate results.
- Encountering words that are not defined in the data set.
- Only works on mobile app.
- Data Accuracy.
- User Knowledge and Training, assume user should know how to use system.
- Label information, content of the product label is assumed to be accurate.
- Users cannot change the content information of foods in the system.

3. Specific Requirements

3.1 Interface Requirements

3.1.1 User Interface

One of the major purposes of our application is that the user interface will be easy to use for each age group, Android and IOS platforms. For this purpose, colors that will tire the eyes and readable fonts can be used. Mobile App tools and UI tools like Figma, inVision, Adobe XD and Sketch can help the Celiac Detection System.

3.1.2 Hardware Interface

No hardware interfaces are needed to run this software.

3.1.3 Software Interface

Java programming language and development tools such as Android Studio can be used when creating mobile applications. Flutter can be used to develop mobile applications that are supported on both IOS and Android platforms. Python or Java languages can be used when developing deep learning or machine learning models.

3.1.4 Communication Interfaces

User Interface Database Interface Command-Line Interface (CLI) Inter-process Communication (IPC) User Input Interfaces Email APIs Data Transfer Interface Output Interface Inter process communication Operating System APIs

3.2 Detailed Description of Functional Requirements

1. Report a problem

Purpose: This function aims to let users report a problem within the application. It requires a type string report and a unique type string username. The system makes sure the user exists and is valid and it will let the user enter the problem into a report box that holds a string value. After

the user correctly provides this information to the system, the system will send this object to the database to save it.

Required Inputs: A unique username(string), a report text(string)

Optional Inputs: none

Input Sources: keyboard

Validity Check: The user should provide the system with correct and valid information. For the report text, it should be non-empty. For the user, they should be logged in to the application and have a valid username that is recognized by the system.

Processing: The system needs to check the validity of the user to save the report into its database. Therefore, the user should be logged in and have a unique, recognized account and provide the report text properly. As long as the user is valid, and they provide the report text then the report will be created and saved correctly.

Expected Error Scenarios: The user may not be logged in or have provided the system with correct or valid information therefore the report object will not be created and saved to the database.

How to Handle Errors: The system will output an error message that explains why the report is not correctly created and what the user should do reflectively.

Outputs: A pop-up message that prints the report is successfully created and submitted. A report object that is a string.

2. Scanning the product

Purpose: It determines whether the product contains gluten.

Description: The product is scanned by the user. It is determined whether the product contains gluten, and this information is printed on the screen.

Input: Image of the product

Processing: After the product is scanned, product information is fetched from the database or external sources. It is checked whether the product contains gluten, and the information is printed on the screen.

Expected Error Scenarios / How to Handle Errors:

- The camera cannot work properly.
- The system may not connect to the database or external sources, in which case it will send an error message.

Output: The screen displays whether the product contains gluten.

3. Voice of the result

Purpose / Description: The result of the product scan (whether the product contains gluten) is voiced.
Input: Text (string), audio (mp3 format)
Processing: After the scanning process is completed, the product information is accessed, and it is checked whether it contains gluten. The result is voiced.
Expected Error Scenarios / How to Handle Errors:

- The system may not connect to the database or external sources, in which case it will send an error message.
- Product information may not be obtained.
- There may be an error in the voice acting process.

Output: The result of whether the product contains gluten is voiced.

4. Save Users to System Database

Purpose: To enable users to access the gluten browser system, this requirement aims to guarantee that user registration data is accurately and securely saved to the system's database during the registration process.
Required Inputs: Registration information submitted by the user, such as password, email address, and username. Any additional user information required during registration.
Detailed Description of Creation of The Input Validity Check:
The system shall verify that the user has provided a valid and unique username. The email address must be verified by the system to ensure that it is correct and has not been registered before. Passwords should meet specified security criteria, such as minimum length, complexity, and no easily guessable patterns.
Expected Error Scenarios:
Database Connection Error:
If there is a failure to establish a connection with the database, the system shall raise a database connection error.
Error Message: "An error occurred while processing your request. Please try again later."

How to Handle Errors: When errors occur, the system shall display clear and user-friendly error messages. Users will be prompted to correct the specific issue and resubmit the registration form. The system shall provide a link or button for users to reset their password in case they forget it. In the case of a database connection error, the system should log the error for system administrators to investigate.

Output: Following successful registration, the system will safely store the user's data in the database. After successfully registering, the user should receive a confirmation message along with any additional instructions, if any.

5. Change password

Purpose: The purpose of this requirement is to provide users with the ability to change their existing password securely within the system.

Required Inputs: User's current password. User's new password.

Detailed Description of Creation of The Input Validity Check:

- To confirm the user's identity, the system will ask them to input their current password.
- The system shall enforce password security criteria for the new password, such as a minimum length, complexity, and no easily guessable patterns.
- The system shall confirm that the new password is not the same as the current password.

Expected Error Scenarios:

1. Incorrect Current Password:

If the user's entered current password does not match the one in the system, an error should be raised.

Error Message: "Incorrect current password. Please try again."

2. Weak New Password:

The system should raise an error if the newly entered password does not match the security requirements.

Error Message: "New password is too weak. It must meet the minimum-security requirements."

3. Same New and Current Password:

If the user attempts to change their password to the same password as the current one, the system should raise an error. Error Message: "New password must be different from the current password."

4. Database Connection Error:

The system ought to raise a database connection error if the attempt is made to update the password without successfully connecting to the database.

Error Message: "An error occurred while processing your request. Please try again later."

How to Handle Errors: When errors occur, the system should display clear and user-friendly error messages. Users should be prompted to correct the specific issue and resubmit the password change form. In the case of a database connection error, the system should log the error for system administrators to investigate.

Output: The user's password should be safely updated in the database by the system following a successful password change. The user should receive a confirmation message indicating the successful password change and any further instructions if needed.

6. Verification

Purpose: After registration operations, the system must send a verification email to the users' email, to verify their accounts. With this way, users will be informed about successful registration and changed personal information by the users from the system with the "My Account" button.

Required Inputs: Users' email address.

Detailed Description of Creation of The Input Validity Check:

- To confirm the users' identity, the system will send them an email that includes a randomly created number with 6 digits.
- The system must compare the input written on the system and the sent number sent in the email to confirm the operation.
- The system must confirm the operation after comparing input written on the system and the sent number that is in the email.

Expected Error Scenarios:

- Wrong Verification Code:

The user may enter the verification code wrong.
Error Message: "The Verification Code is Wrong! Please try again.".

- Wrong email address:

The user may enter the email address wrong.
Error Message: “Invalid email address. Please try again.”.

- **Verification Code error:**

System may not send verification code correctly to the users’ email.
Error Message: “Something went wrong. Please try again”.

- 4. Database Connection Error:

Database connection problems may occur during these operations.
Error Message: “Something went wrong. Please try again”.
How to Handle Errors: When errors occur, the system should resend a new verification code to the users’ email. Users should be prompted to correct the specific issue and resubmit the operation. In the case of a database connection error, the system should log the error for system administrators to investigate.
Output: The users’ registration and information changes should be saved into the database correctly.

7. Allowance

Purpose: To analyze the input picture, the system must have permission to use the phone camera.
Required Inputs: Users’ permission.
Detailed Description of Creation of The Input Validity Check:

- At the first time of users’ usage of application, to allow the usage of the camera, the system sends a notification from the phone screen about the phone camera, after clicking the “Scanning” button,
- The users must click “Allow” to give permission to the system for the scanning operation.

- The system must open the phone camera if the users click “Allow”, otherwise the system must return to the main menu.
- The system must send notification again to get an allowance whenever they click the “Scanning” button, if the users refuse to allow the camera usage.

Expected Error Scenarios:

- **Notification error:**

The system may not send the notification error to users.
Error Message: "Something went Wrong! Please try again.".

- **Camera error:**

The system may not open the phone camera correctly.
Error Message: "Something went Wrong! Please try again.".

- **Database Connection Error:**

Database connection problems may occur during these operations.
Error Message: “Something went wrong. Please try again”.
How to Handle Errors: In the case of a database connection error, the system should log the error for system administrators to investigate.
Output: After getting permission to use the camera, the system opens the camera for the scanning operations.

8. Log In

Purpose / Description: This function authenticates users based on the information they enter into the system and allows them to access the system.
Required Inputs: Username (strings, user’s unique identifier), Password (string, a secret key for authentication)
Optional Inputs: None
Input Sources: User Interface (username and password are entered by the user on the phone keyboard)
Validity Check: Username and password fields must be filled in, verify that the username exists in the database, verify that username and password match.

Processing: Receive inputs (username and password) from the user. Check that all input fields are filled and valid. Ask the user database to verify if the entered username exists. Compare the password entered as input with the password for the username in the database. Verify access to the system and display a success message on the screen.

Expected Error Scenarios:

1. Invalid Input:

- Username or password information not entered.
- The username and password do not match.
- Inputs do not exist in the database.

2. Database Connection Error:

- Unable to connect to system database.

How To Handle Errors:

1. Invalid Input:

- An error notice stating that the login credentials are invalid is shown by the system.
- The system allows the user to reset the password, or the option forgot username.

2. Database Connection Error:

- The system displays an error message indicating a problem with the connection to the database.
- Suggest try again or report a problem option.

Outputs:

1. Success:

- Display a success message indicating that the sign in was successful.

2. Failure:

- **Invalid Input:** Display an error message indicating that the user cannot sign in to the system.
- **Database Connection Error:** Display an error message indicating that the system cannot connect to the database.

9. Register

Purpose / Description: The registration process allows new users to create an account and gain access to the system's features.

Required Inputs: User details, including but not limited to username, email address, and password.

Description:

- The system validates the entered information to ensure accuracy and completeness.
- Passwords are securely hashed and stored.
- Email verification is sent to the provided email address.

Expected Error Scenarios:

1. Duplicate Username:

The system will raise an error if the username you entered is already in the database.

Error Message: "Username already in use. Please choose a different username."

2. Invalid Email Format:

An error message will be displayed by the system if the email address entered is not in a valid email format.

Error Message: "Invalid email address format. Please enter a valid email address."

3. Duplicate Email Address:

If the entered email address is already registered, the system shall raise an error.

Error Message: "This email address is already associated with an account. Please use a different email address."

4. Weak Password:

An error message will appear on the screen if the password you entered does not match the security requirements.

Error Message: "Password is too weak. It must meet the minimum-security requirements."

Output: Upon successful registration, the system creates a new user account. An email verification process is initiated, and the user is prompted to confirm their registration by clicking on the verification link.

10. Edit Profile

Purpose / Description: The Edit Profile functionality allows users to modify and update their profile information within the system.
Required Inputs: User details such as username, email address, and other customizable profile information.
Description:

- The system should validate and update the user's information, ensuring data accuracy.
- Changes made by the user should be reflected in the user profile.

Expected Error Scenarios / How to Handle Errors:

- The system should handle scenarios where the user attempts to save incomplete or invalid information.
- In the case of server errors or database connection issues during the update process, the system should display a relevant error message.
- If the user attempts to use an email address or username that is already in use, the system should prompt the user to choose a different one.

Output: Upon successful validation, the system updates and saves the user's profile information. An error message is displayed if there are issues with the update process.

3.3 Non-Functional Requirements

1. Performance

- The application's performance requirement dictates that the time taken from the scanning of a barcode or QR code to the display of the gluten information must not exceed 3 seconds. This duration includes both the image processing time and the backend database query time.
- The performance benchmark under high traffic conditions specifies that the application should maintain its responsiveness and quick processing time without any visible lag or delay, even when faced with a significant user load.

2. Reliability

- The system must deliver accurate and reliable results by utilizing correct information from the database.
- Regular backups and data integrity checks must be conducted to ensure the reliability of information stored in the database.

3. Usability

- The usability requirement emphasizes the need for an intuitive and user-friendly interface, minimizing the need for extensive user training and ensuring that users can effortlessly navigate the application's features and functions.
- The application's adaptability to various screen sizes and resolutions should allow for a seamless and consistent user experience across a wide range of mobile devices.

4. Security

- The security standard requires that all user data, including personal information and preferences, must be encrypted using advanced encryption algorithms, ensuring data confidentiality and integrity throughout the system's operations.
- The application's security protocol should include regular security audits, vulnerability assessments, and timely updates to safeguard the system against potential security breaches and data leaks.

5. Scalability and Performance

- The scalability requirement mandates that the application should be capable of accommodating an increasing number of concurrent users without compromising the speed and efficiency of the gluten information retrieval process.
- The database's performance should remain consistent and efficient, even when handling large datasets and experiencing fluctuations in user traffic, ensuring that the application's responsiveness is not compromised under heavy loads.

6. Compatibility

- The compatibility standard dictates that the application should function seamlessly across various popular mobile operating systems, including but not limited to Android 7.0 and iOS 11, ensuring a wide user base and accessibility for individuals with different device preferences.
- The barcode or QR code scanning functionality should be compatible with a diverse range of formats commonly used in the packaging of food products, guaranteeing comprehensive coverage and reliable information retrieval for all scanned items.

7. Performance Monitoring

- The performance monitoring requirement necessitates the continuous monitoring of the application's response time, system errors, and user feedback to identify potential performance bottlenecks and ensure prompt troubleshooting and maintenance.
- An efficient performance monitoring system should provide real-time insights into the application's resource utilization, allowing for proactive measures to optimize the application's performance and enhance the user experience.

8. Mobile Power Usage

- The mobile power usage specification aims to minimize the application's energy consumption during barcode and QR code scanning operations, optimizing the device's battery life and reducing the overall power drain.
- By implementing power-saving modes and efficient resource management strategies, the application should ensure that the camera usage does not significantly impact the device's battery life, allowing users to utilize the application for extended periods without experiencing rapid battery depletion.

4. Analysis - UML

4.1 Use Cases

1. Use Case: Report a Problem

Use Case Number: UC1

Actor: User

Overview: A user wants to report a problem within the system. They need to enter their report text and be logged in to report a problem.

Typical Flow Description (Primary Scenario):

Precondition:

- User has an account.
- User is logged in.
- User fills in the report form.

Postcondition:

- User successfully filled in the report form and submitted the report into the system.
- User encounters a problem.
- They click on the submit a report button.
- They enter their report into the report form.
- Entered values are accepted and the user is valid, the system responds with a message saying the submission is successful.
- The system saves the user report.

Alternative Flow Description (Secondary Scenario):

Precondition:

- User does not have an account.
- User does not have a valid internet connection.
- User is not logged in.
- Users did not fill in the report form as they should.

Postcondition:

- Users cannot report a problem.
- The system cannot connect to the database.
- User lost the internet connection while trying to submit the report form.
- User was not logged in before trying to submit the report form.

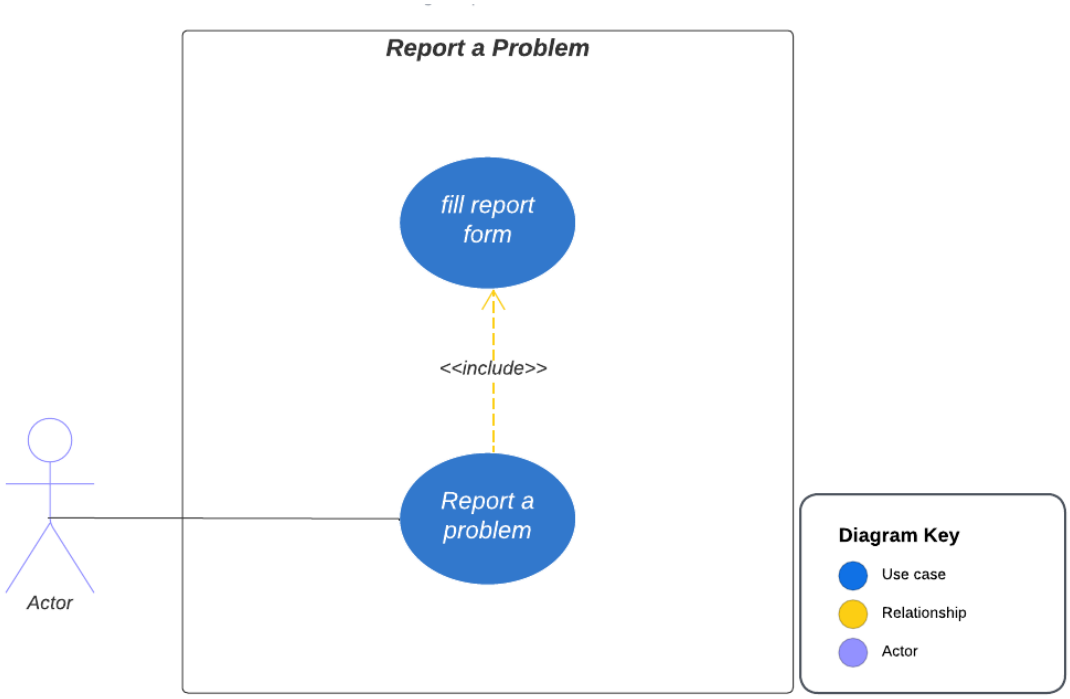

Figure 2: Report a Problem Use Case

2. Use Case: Scanning the product

Use Case Number: UC2

Actor: User

Overview: The user scans the products to see if they contain gluten.

Related use cases: UC7, UC8

Typical Flow Description (Primary Scenario):

Precondition:

- The user logged in to the system.
- The user must give permission to use the camera.

Postcondition:

- The screen prints whether the product contains gluten.

1. The user clicks the button called scan on the screen.
2. Users scan the product with the camera.
3. The product information is retrieved from a database or external sources.
4. The system checks whether the product contains gluten based on the product information.
5. The screen displays whether the product contains gluten.

Alternative Flow Description (Secondary Scenario):

Precondition:

- The user logged in to the system.
- The user must give permission to use the camera.

Postcondition:

- The user cannot scan the product properly.
- The system displays an error message.
- The user cannot tell whether there is gluten in the ingredients of the product.

2.a. The camera cannot work properly.

2.b. The writings on the product cannot be readable.

3.a. Product information may not be retrieved from the system.

5.a. If the product content cannot be reached, the system displays an error message.

Figure 3: Scanning the Product Use Case

3. Use Case: Voice of the result

Use Case Number: UC3

Actor: User

Overview: The result of the product scan (whether the product contains gluten) is voiced.

Related use cases: UC2, UC7, UC8

Typical Flow Description (Primary Scenario):

Precondition:

- The user logged in to the system.
- The user must give permission to use the camera.

Postcondition:

- The result is voiced.

1.User scans the product with the camera.

2.The user clicks the button called voice on the screen.

3.The result of whether the product contains gluten is voiced.

Alternative Flow Description (Secondary Scenario):

Precondition:

- The user logged in to the system.

- Users scan the product with the camera.

Postcondition:

- If there is a problem with the scan result, the result cannot be voiced.
- The system displays an error message.

1.a. Product information may not be obtained.

3.a. There may be an error in the voice acting process.

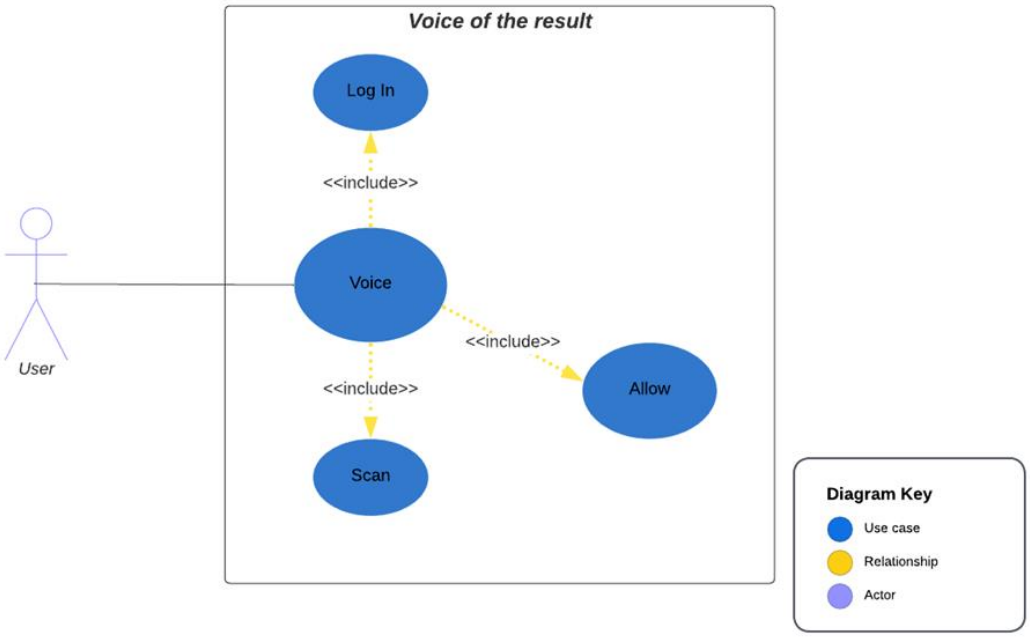

Figure 4: Voice of the Result Use Case

4. Use Case: Save Users to System Database

Use Case Name: Save Users to System Database

Use Case Number: UC4

Actor: User, System

Overview: This use case outlines the process of securely registering and storing user data within the system's database.

Related Use Cases: UC1

Typical Flow Description (Primary Scenario):

Preconditions:

- The user has access to the registration form.
- The system is operational and connected to the database.

Postconditions:

- User data is securely stored in the system's database.
- The user receives confirmation of successful registration.

1. Accesses the registration form
2. Displays the registration form containing fields for username, email address, password, and necessary details.
3. Users enter their registration details.
4. Securely stores the user's registration information in the database upon successful validation, generating a distinct user ID.
5. Authenticates the user for access to system features.
6. Displays a confirmation message indicating successful registration.

Alternative Flow Description (Secondary Scenario):

Preconditions:

- The user has access to the registration form.
- The system encounters a database connection error.

Postconditions:

- Users are notified of the technical issue and advised to attempt the registration process later.
- Informs the user of the technical issue due to a failure to connect to the database.

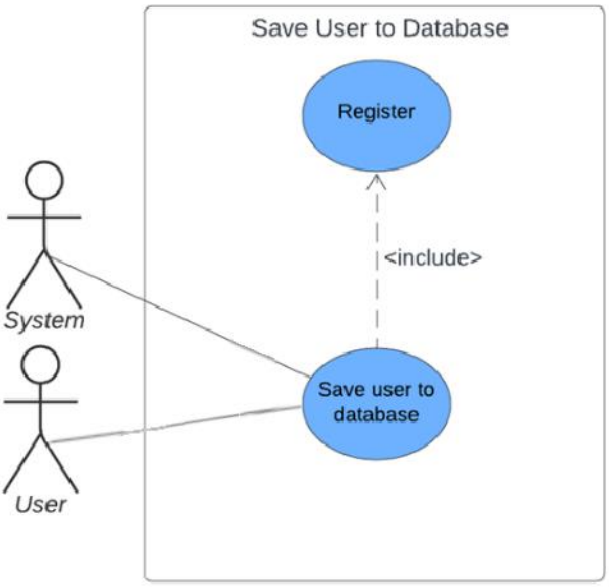

Figure 5: Save User to Database Use Case

5. Use Case: Change Password

Use Case Number: UC5

Actor: User

Overview: The user wants to change their password.

Related use cases: UC1

Typical Flow Description (Primary Scenario):

Precondition:

- The user entered the system's login information.
- The user is aware of their present password.
- The system allows users to change their current password.

Postcondition:

- The user's password successfully updated in the system and database.
- The user can log into their account with the updated password.

1. The user selects the change password option.
2. The system wants the user to enter their current password.
3. The user enters.
4. The system checks the password and validates it.

5. To input new data, the user is presented with an interface like a form where the system prompts them to enter a new password and make sure it works.
6. The user enters a new password and then submits it.
7. The system checks the new password and if it is valid the system updates the database with the new password.
8. A confirmation message is shown by the system on the screen, which shows that the operation is successfully completed.

Alternative Flow Description (Secondary Scenario):

Precondition:

- The user logged in to the system.
- The user knows their current password.
- The system allows users to change their current password.

Postcondition:

- The user's password stays unchanged.
- The updated password could not be saved by the system.
- The system displays an error message.
- The system requests that the user update by entering a valid password again.

4.a. An invalid password was entered.

7.a. The latest entered password and confirmation password do not match with each other.

7.b. The newly entered password does not meet the password specifications.

7.c. An error with the database connection or the system failing to update.

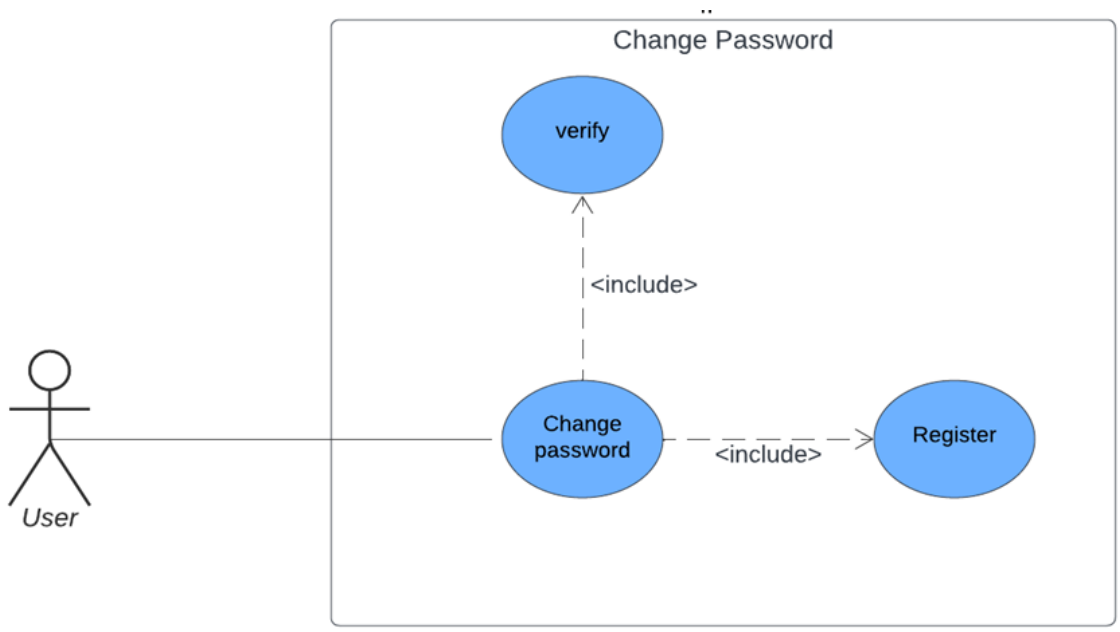

Figure 6: Change Password Use Case

6.Use Case: Verification

Use Case Number: UC6

Actor: User

Overview: A user must verify his/her account for the sign-in process.

Related use cases:

Typical Flow Description (Primary Scenario):

Precondition:

- The user writes the wrong verification code into the system.

Postcondition:

- System warns the user about the wrong verification code and goes back to the registration page.
- The user fills the registration page for his account.
- The system sends a verification code to the user's email.
- The user enters the verification code into the system to verify his/her account.
    - The user presses the “Verify” Button to save his/her account into the system.

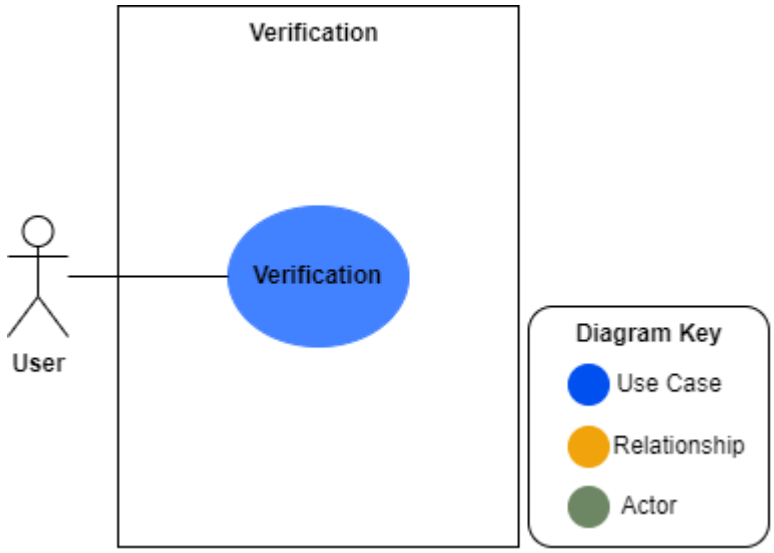

Figure 7: Verification Use Case

7. Use Case: Allowance

Use Case Number: UC7

Actor: User

Overview: A user must allow the camera usage of the application for the scanning operations.

Related use cases: UC2 - UC3 - UC8

Typical Flow Description (Primary Scenario):

Precondition:

Users did not allow camera usage.

Postcondition:

- The scan operation cannot be performed.
- User presses the scanning button.
- After pressing the button, the user is informed by the system for the usage of the mobile phone camera.
- If the user allows the system to use the camera, the scanning screen can be seen, otherwise the system returns to the main menu.

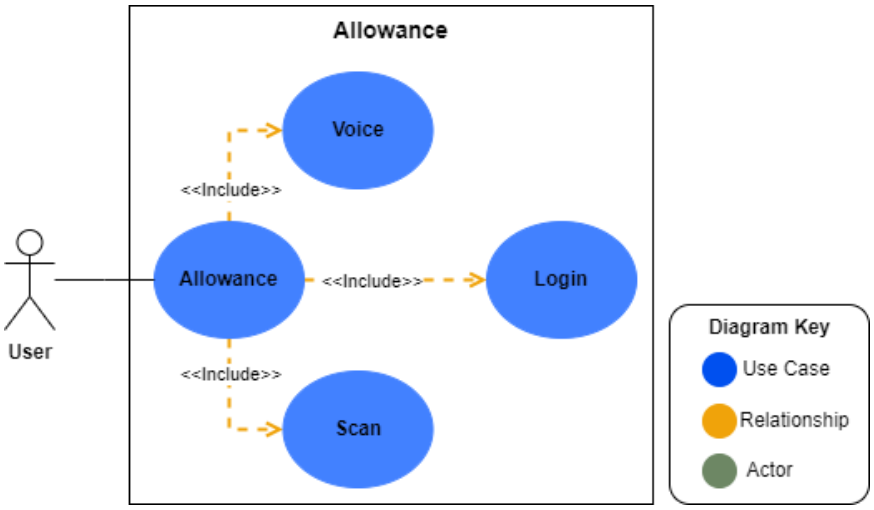

Figure 8: Allowance Use Case

8.Use Case: Login

Use Case Number: UC8

Actor: User

Overview: A user wants to log in to the system. They need to enter her/his username and password to enter the system.

Related use cases: UC9

Typical Flow Description (Primary Scenario):

Precondition:

- User has an account.
- User has an internet connection.
- User enters a valid username or email address and the correct password to log in.

Postcondition:

- User successfully logged in to her/his account and can access her/his accounts' all properties.

1. The user opens the login screen, and the system displays the corresponding screen asking the user to enter their username and password.
2. User enters her/his username and password correctly into the related field on the screen.
3. The entered information is accepted and matched, the system responds with authentication on the user.
4. System displays the user's account.

Alternative Flow Description (Secondary Scenario):

Precondition:

- User does not have an account.
- User does not have an internet connection.
- User does not remember her/his username and password.

Postcondition:

- User cannot connection access her/his account

1. The system cannot connect to the database.
2. Any technical failure occurs in the system.
3. Internet connection is interrupted while the user is trying to sign in.
4. User's account has been deleted and therefore the account no longer exists.

Figure 9: Log in Use Case

9.Use Case: Register

Use Case Number: UC9

Actor: User

Overview: A user wants to register to the application. They need to enter their username and password to enter.

Related use cases: UC6

Typical Flow Description (Primary Scenario):

Precondition:

- The user has downloaded and installed the application.
- User has no account in this application before.
- Users should type the information that the system asks.
- Users shouldn’t be registered for the system before.

Postcondition:

- The user is successfully authenticated and granted access to the application's main interface.

1. The user launches the application.
2. The application displays the entrance screen prompting the user to enter their username and password or register.
3. The user enters the mail, username, and password.
4. The user clicks the "Register" button.
5. The application verifies the user's credentials.
6. If the credentials are valid, the application grants access to the main interface.
7. The user can now utilize the application's features and functionalities.

Alternative Flow Description (Secondary Scenario):

Precondition:

- User does not have an account.
- User does not have a valid internet connection.
- User forgets the password if the user already exists in the system.
- Users did not fill in the asking information as they should.

Postcondition:

- If the user fails to provide valid credentials, the application displays an error message.
- (Step 5 Alternative) If the user provides incorrect credentials, the application displays an error message indicating that the username or password is invalid.
- (Step 4 Alternative) The user can attempt to register again by re-entering the correct credentials.

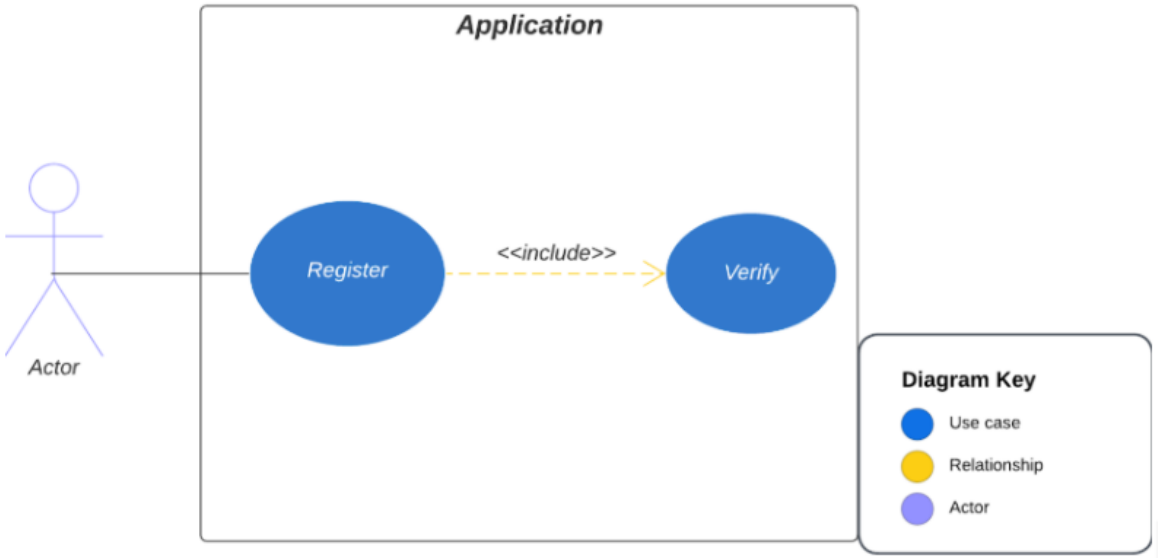

Figure 10: Register Use Case

10. Use Case: Edit Profile

Use Case Name: Edit Profile

Use Case Number: UC10

Actor: User

Overview: This use case enables users to modify and update their profile information within the application.

Related Use Cases:

Typical Flow Description (Primary Scenario):

Precondition:

- The user is logged in to the application.

Postcondition:

- The user's profile information is successfully updated and saved.

- The user navigates to the "Edit Profile" section within the application.
- The application displays the user's current profile information.
- The user makes the necessary changes to the profile details.
- The user clicks the "Save" button to confirm the changes.
- The application validates the updated information.
- If the information is valid, the application updates and saves the user's profile.
- The user is notified that the profile has been successfully updated.

Alternative Flow Description (Secondary Scenario):

Precondition:

- The user is logged in to the application.

Postcondition:

- If there are any issues with the updated information, the application displays an error message.
- (Step 5 Alternative) If the updated information is incomplete or invalid, the application displays an error message indicating the specific issue with the provided data.
- (Step 3 Alternative) The user can revise the changes and reattempt to save the profile information by clicking the "Save" button again.

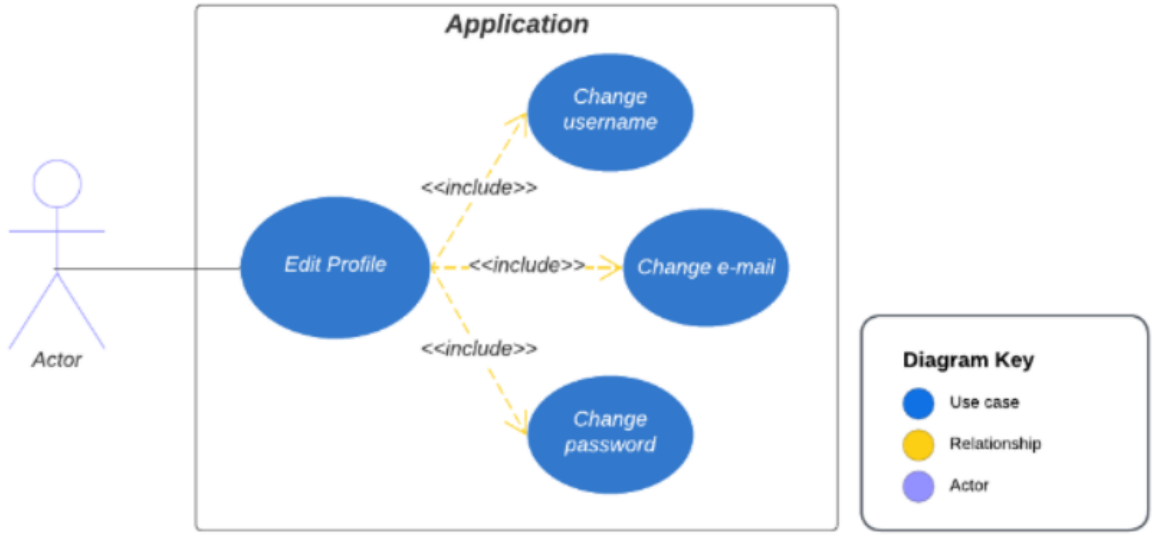

Figure 11: Edit Profile Use Case

11.Use Case: Log Out

Use Case Number: UC11

Actor: User

Overview: The user will be logged out of the system.

Related use cases: UC8

Typical Flow Description (Primary Scenario):

Precondition:

- The user is logged into the system with an account.

Postcondition:

- User logged out of the system.
- User personal data is deleted in the database.
- User redirected to the home page.

1. User selects the “Log Out” option from the system’s user interface.
2. The display shows a confirmation message that the system has been exited.
3. The system closes the session by destroying the user's personal data in the database.
4. The system redirects the user to the home page.
5. The home page is shown to the user.

1. Alternative Flow Description (Secondary Scenario): User cancels sign out.

Precondition:

- User selected log out option.

Postcondition:

- User did not log out of the system.
- User’s personal data is not deleted in database.
- User returns to the home page and remains in the system.

2a. User selects "Cancel" when a confirmation message appears on the screen.

3. User returns to the home page.

2. Alternative Flow Description (Secondary Scenario): Error during sign out.

Precondition:

- User selected log out option.
- An error happened during the log out process.

Postcondition:

- Users cannot log out of the system.
- User's personal data is not deleted in the database.
- System displays an error message.
- User returns to the home page.

3a. An error occurred in the system while logging out the user.

4. System displays an error message.
5. User returns to the home page.

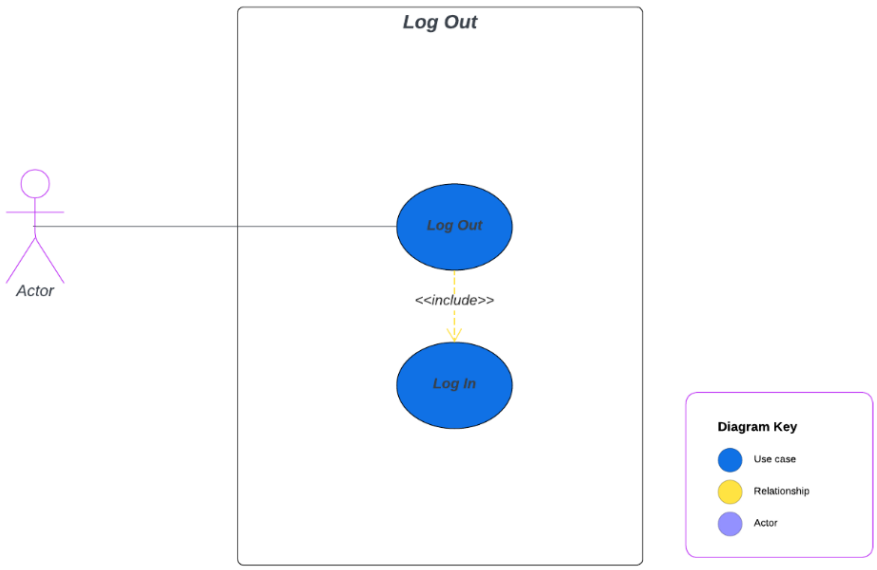

Figure 12: Log Out Use Case

4.2. Functional Modeling

4.2.1. Data Flow Diagrams (DFD)

Level 0

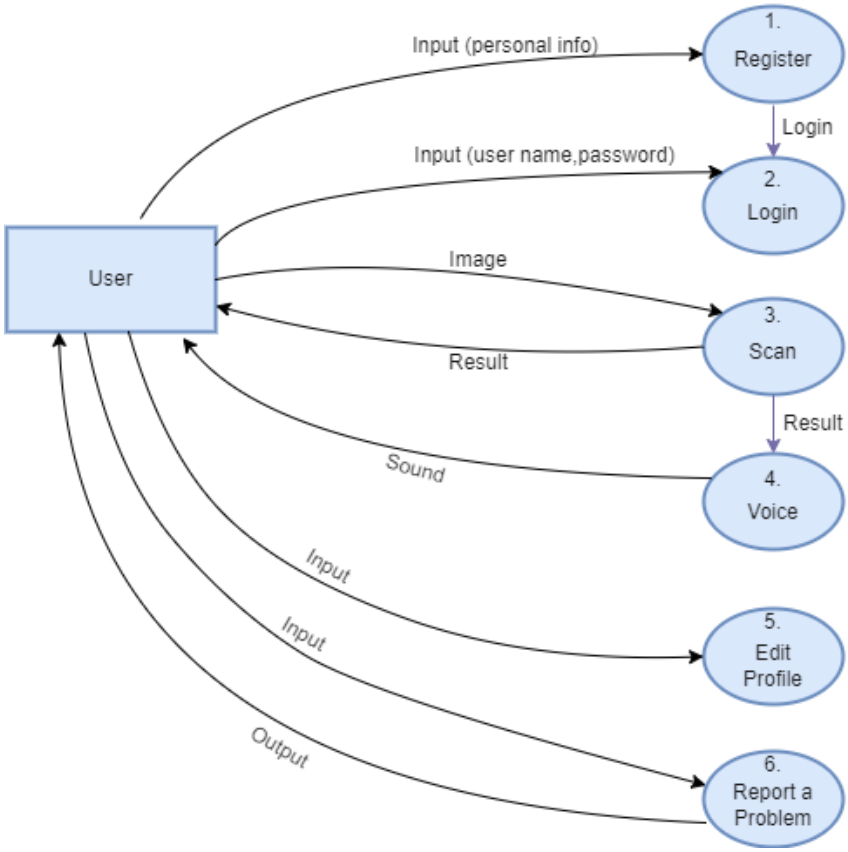

Figure 13: Level 0 Data Flow Diagram

Level 1

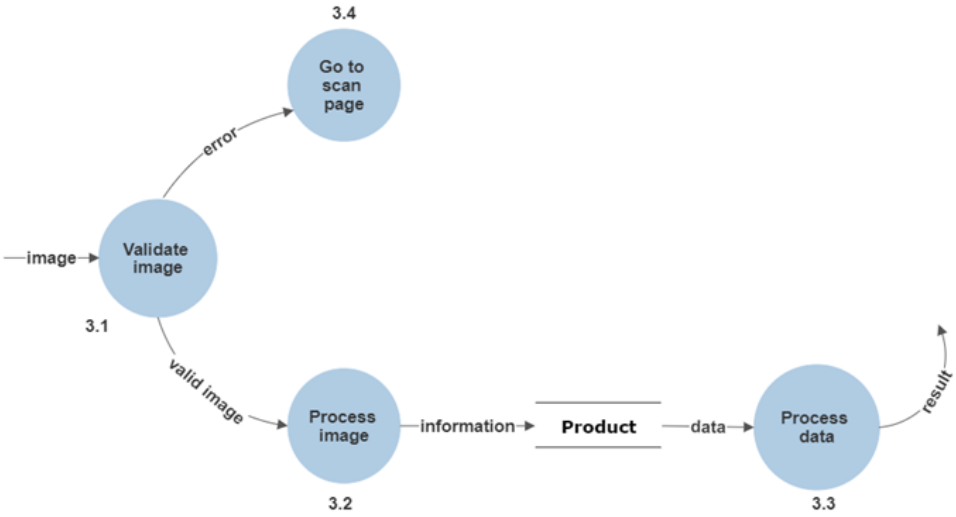

Figure 14: Level 1 Diagram of Scan Operation

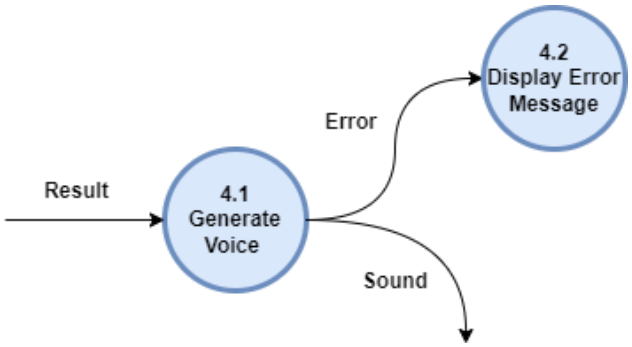

Figure 15: Level 1 Diagram of Voice Operation

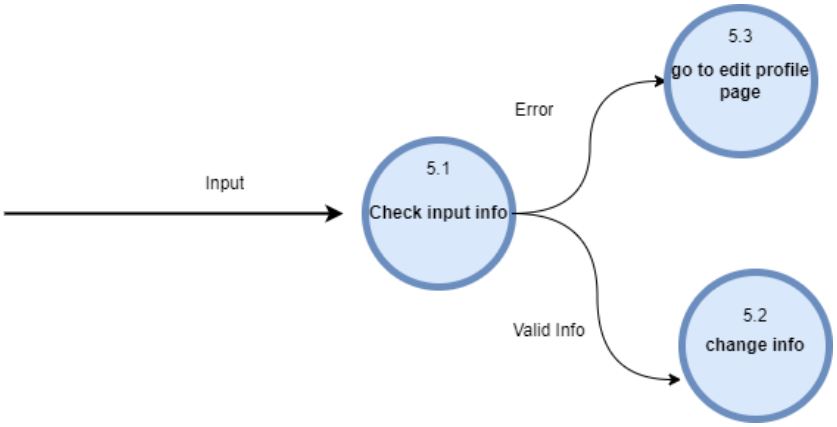

Figure 16: Level 1 Diagram of Edit Profile Operation

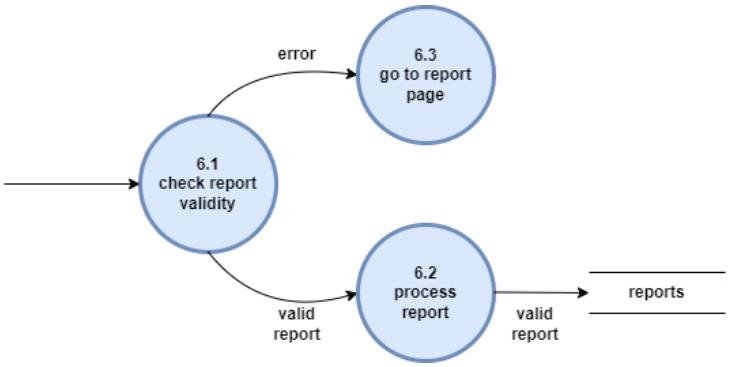

Figure 17: Level 1 Diagram of Report a Problem Operation

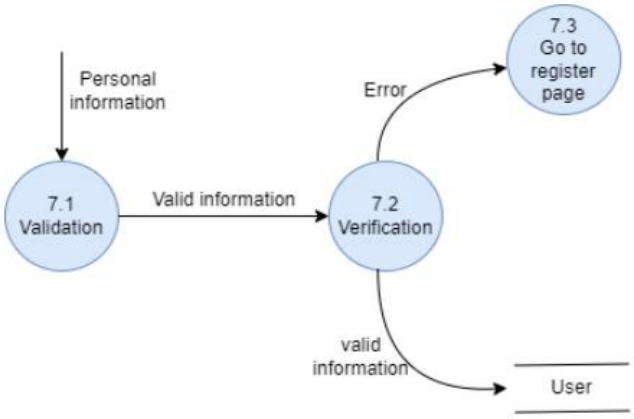

Figure 18: Level 1 Diagram Validation Operation

4.2.2. Data Dictionary

3.1 Dictionary

- image = [valid image, error]
- error: string
- valid image: string
- information: string
- data: string
- result: string

4.1 Dictionary:

- Result: string
- Error: string
- Voice: Media

5.1 Dictionary

- input: string username + email address
- error: string
- valid info: string

6.1 Dictionary

- error: string
- valid report: report input + username
- report input: string
- username: string

7.1 Dictionary

- valid information: username + email password
- personal information: username + email + password
- username: string
- email: string
- password: hash code
- error: string

5. References and Resources

Cankaya University, 07.03.2018, CENG 396, Software Engineering,https://ceng396.cankaya.edu.tr

lucid,06.12.2023,lucidchart,https://www.lucidchart.com

Software Design Document (SDD)

1. Introduction

1.1 Purpose of this document

This software design document describes the architecture and system design of Gluten Detection System. The purpose of this document is to define and explain the concepts and system design. This document aims to make it easier for the people in the development of this project and make it clear to someone who may be joining the project. It also makes sure to document every step of the project so any change that may be made is easier to follow. A SDD is a crucial part of software development.

2. System Overview

Gluten Detection System is a software platform that is designed to simplify the process of identifying gluten in products. This application leverages the power of technology to provide users with an easy and accurate solution for gluten detection. In this platform selected hosts represent their hometown by providing VR tours, online lessons and any information users may request. Users must first register into the system and enter information about what they want to see in their feed, users can filter by location, language, and interests. The system is all about what the users wish to know about, economic, cultural, social information, events, language lessons and many more events. Interface is meant to be easy and simple, suitable for research about specific countries.

6. System Design

3.1 Architectural Design

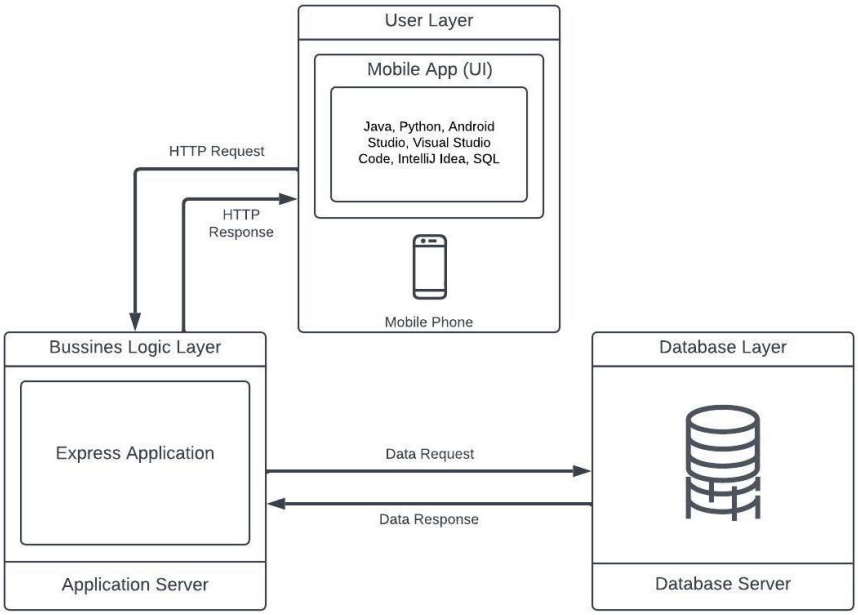

Figure 19: Architectural Design

3.2 Decomposition Description: Class Diagram

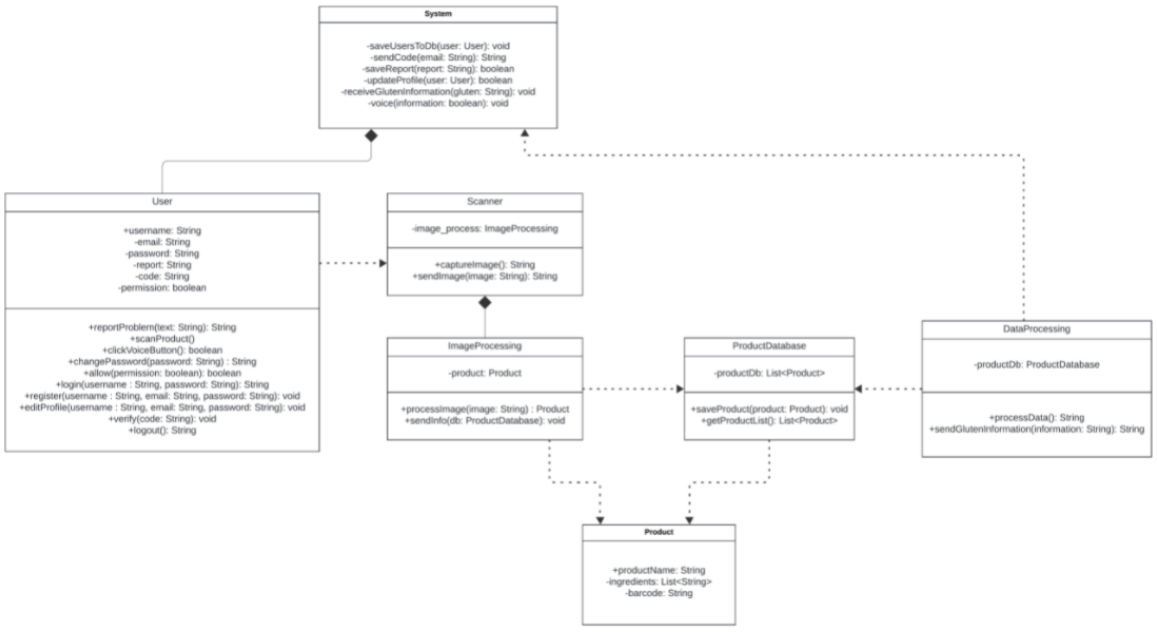

Figure 20: Class Diagram

3.3 System Modelling

3.3.1 Activity Diagrams

1. Change Password

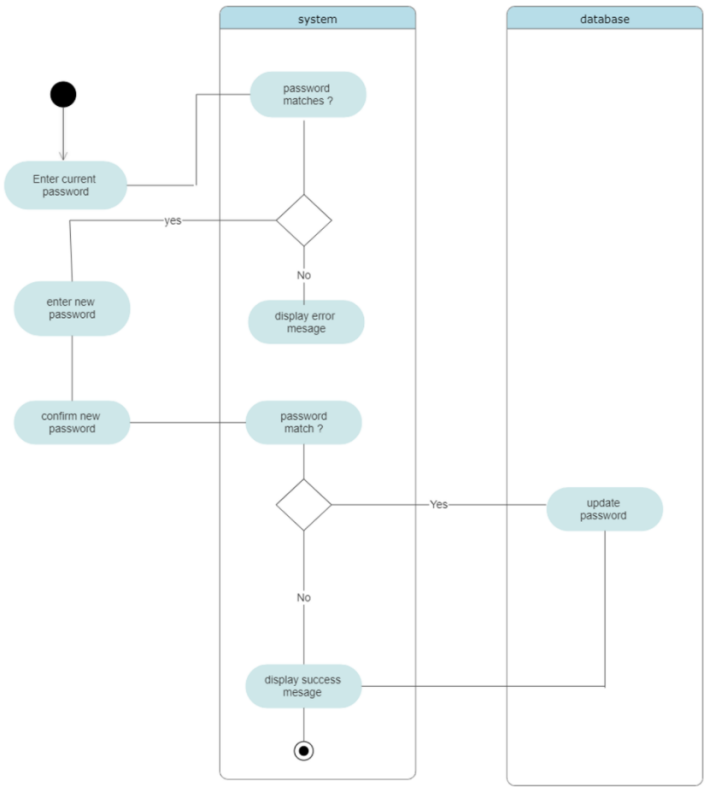

Figure 21: Change Password Activity Diagram

2. Scanning the Product

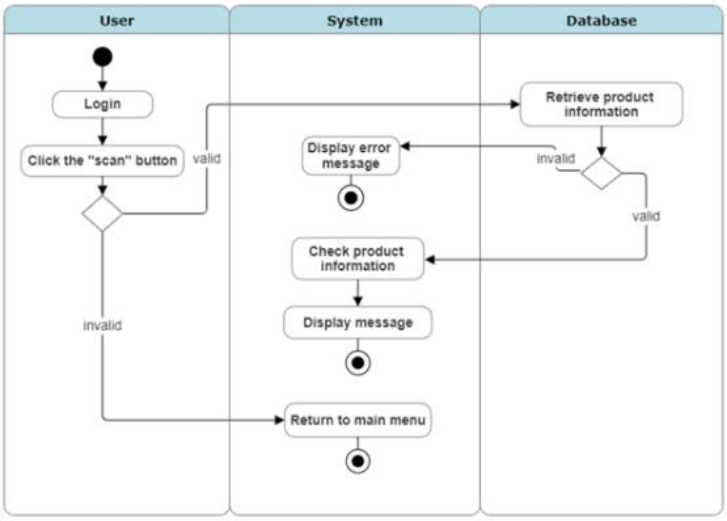

Figure 22: Scanning the Product Activity Diagram

3. Edit Profile

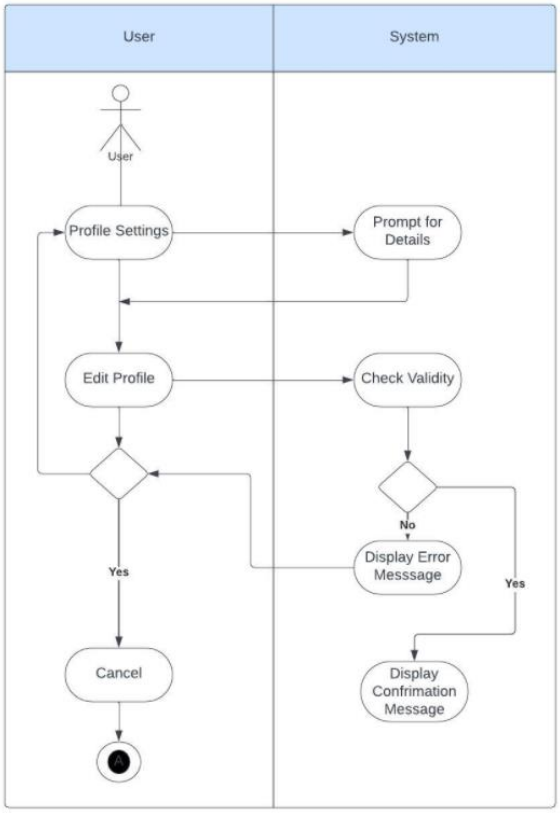

Figure 23: Edit Profile Activity Diagram

4. Verification

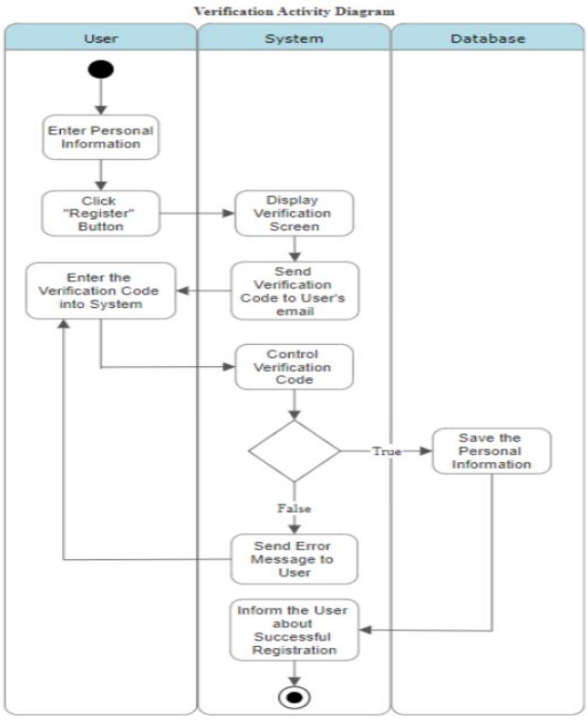

Figure 24: Verification Activity Diagram

5. Allowance

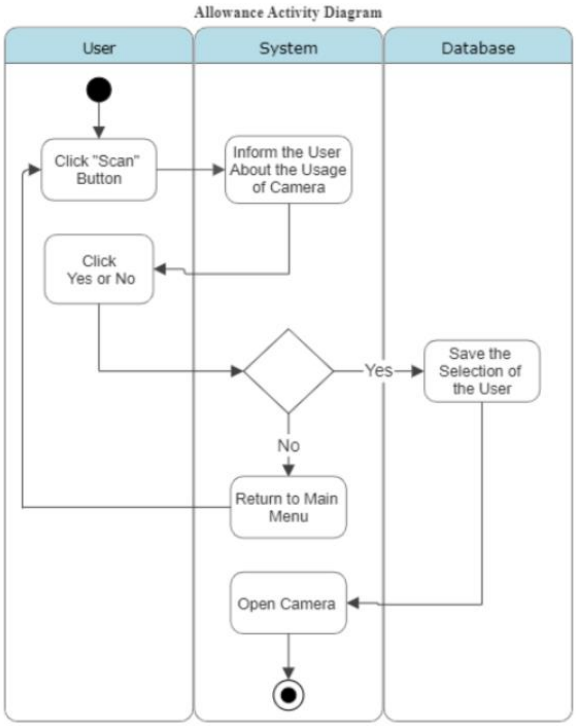

Figure 25: Allowance Activity Diagram

6. Login

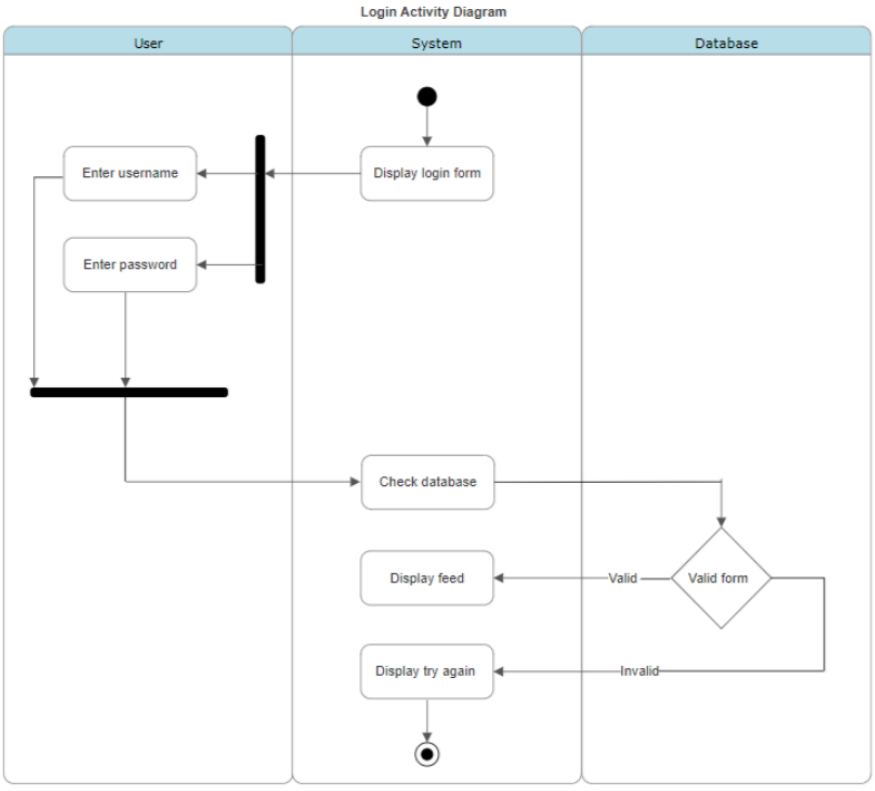

Figure 26: Login Activity Diagram

7. Report a Problem

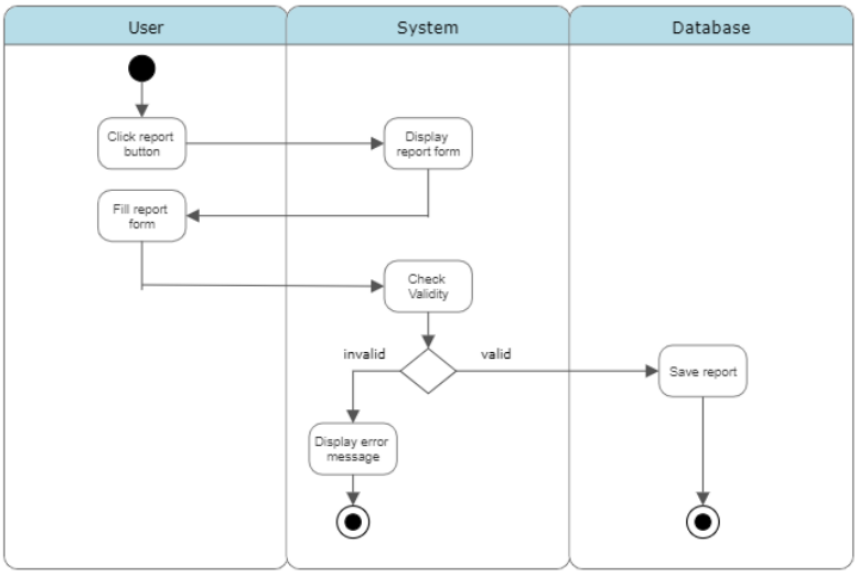

Figure 27: Report a Problem Activity Diagram

3.3.2 Sequence Diagrams

1. Change Password

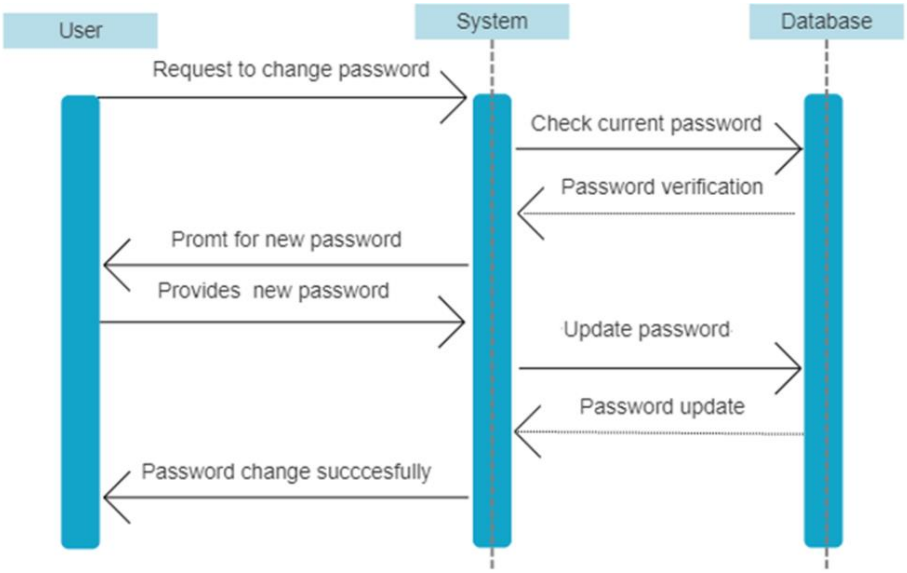

Figure 28: Change Password Sequence Diagram

2. Scanning the Product

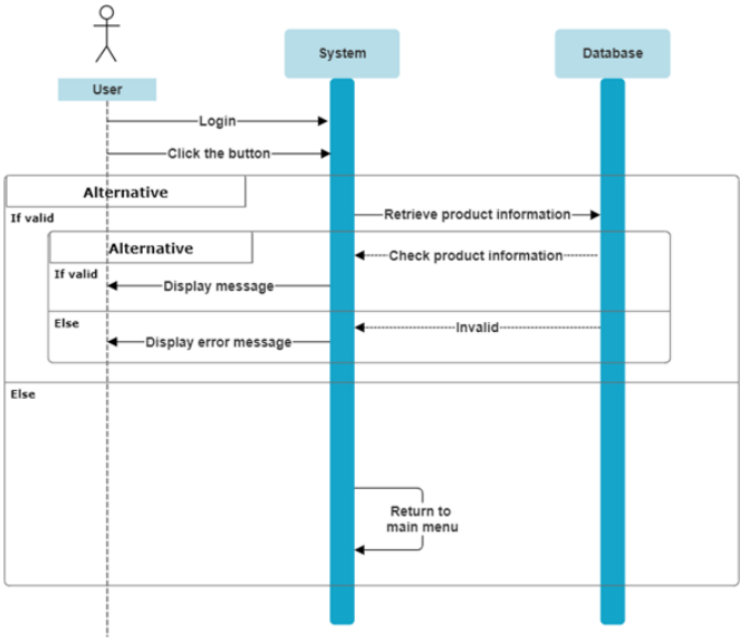

Figure 29: Scanning the Product Sequence Diagram

3. Edit Profile

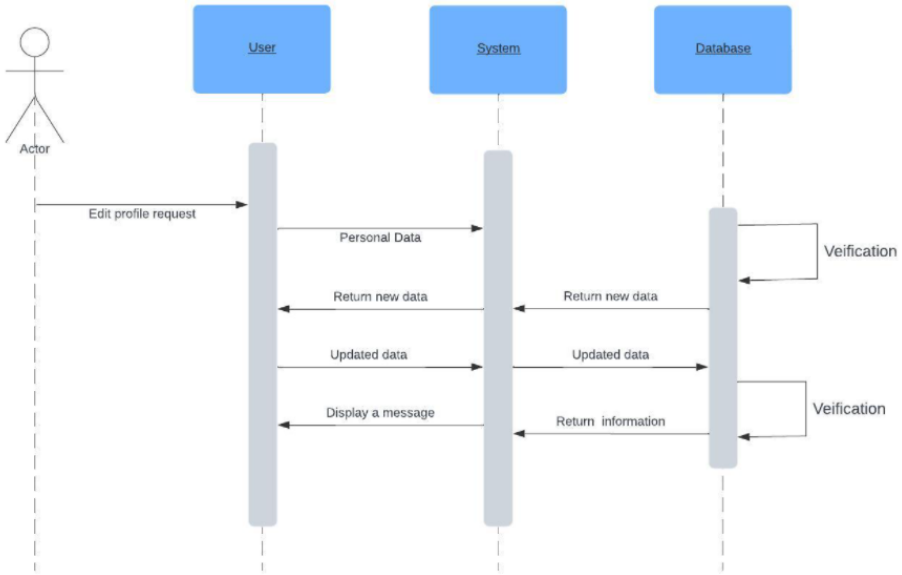

Figure 30: Edit Profile Sequence Diagram

4. Verification

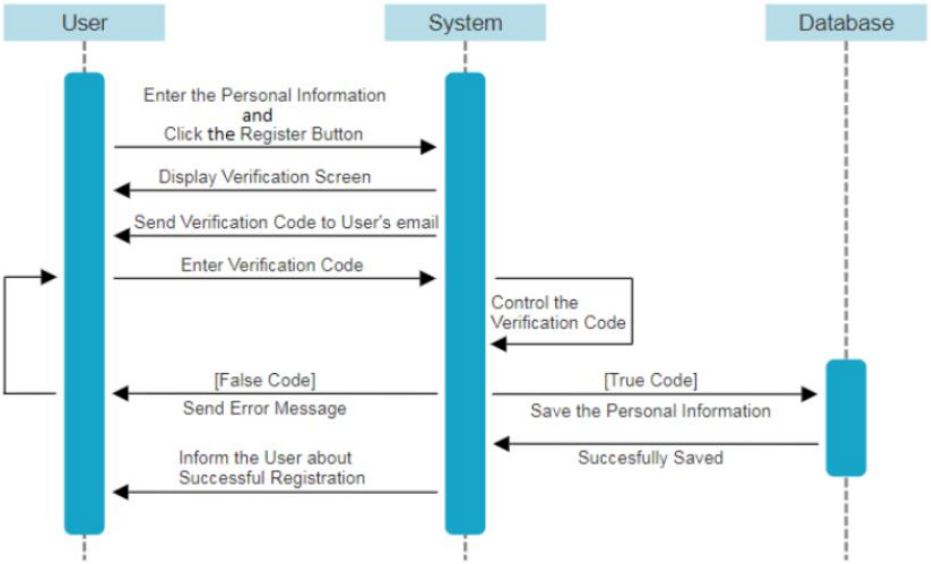

Figure 31: Verification Sequence Diagram

5. Allowance

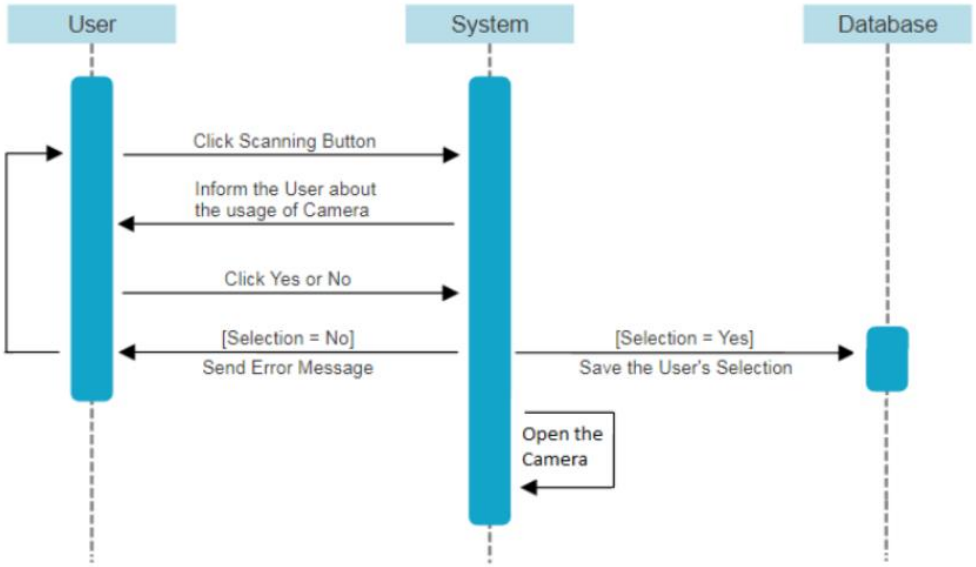

Figure 32: Allowance Sequence Diagram

6. Login

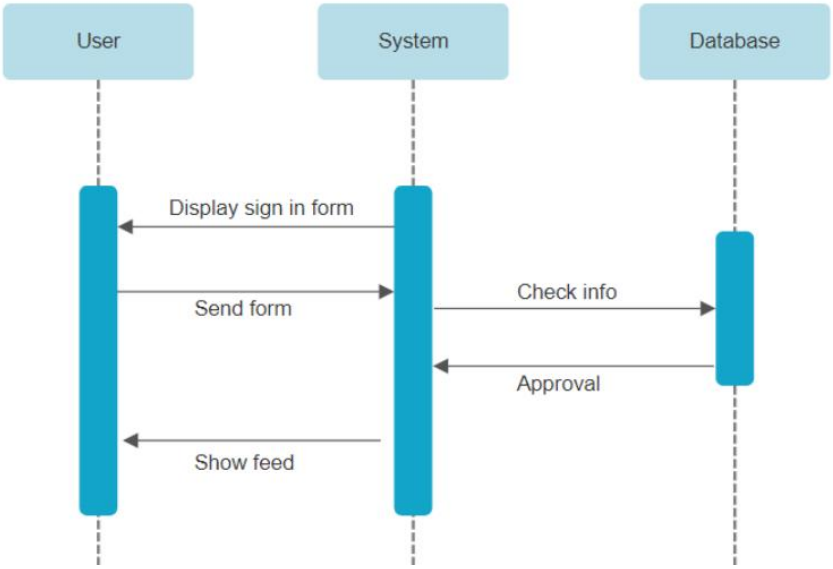

Figure 33: Login Sequence Diagram

7. Report a Problem

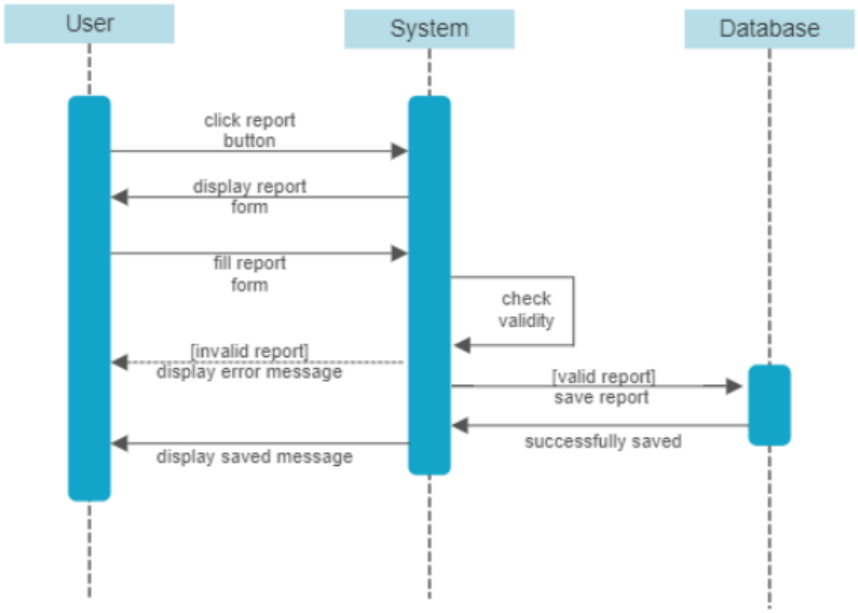

Figure 34: Report a Problem Sequence Diagram

4. User Interface Design

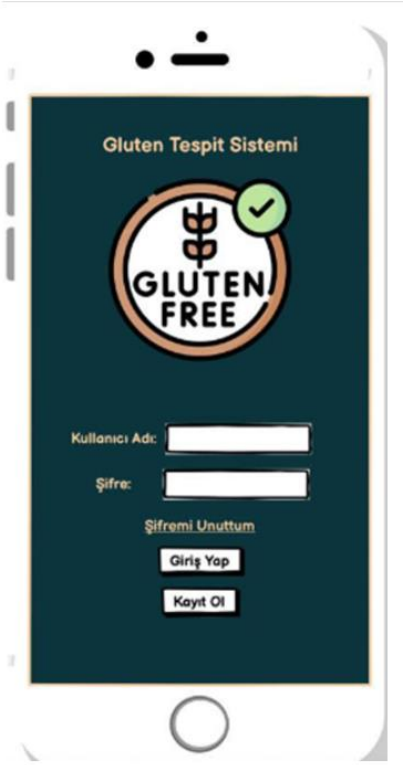

Figure 35: User Interface Screen 1

Figure 36: User Interface Screen 2

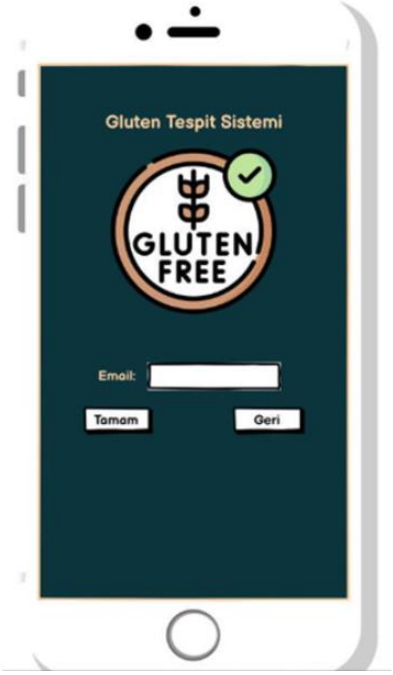

Figure 37: User Interface Screen 3

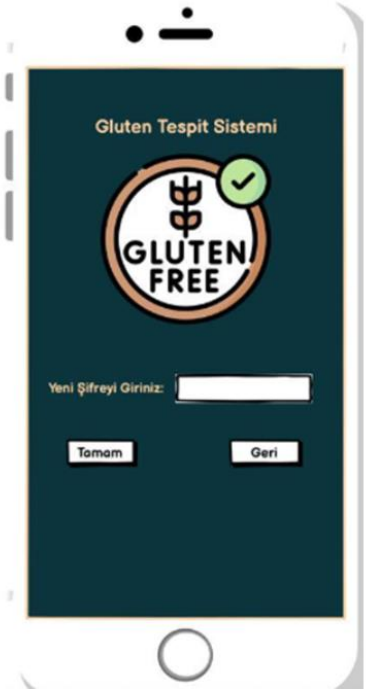

Figure 38: User Interface Screen

Figure 39: User Interface Screen 5

Figure 40: User Interface Screen 6

Figure 41: User Interface Screen 7

Figure 42: User Interface Screen 8

5. Conclusion

The revolutionary software platform created to streamline the process of identifying gluten-containing products is defined by the design and architecture of the Gluten Detection System. The purpose of this article is to clarify the system's architecture and basic principles for both newcomers and the teams working on the project. It highlights how the technology solutions integrated into the system allow users to quickly and effectively identify gluten-containing foods. It also acts as a platform by introducing users to their cities and offering a variety of information through carefully chosen hosts. A strong infrastructure and an intuitive user interface form the foundation of this design. When users register on the system and filter according to their interests, they can quickly obtain the information they want. The Gluten Detection System's design is painstakingly made to guarantee that users can obtain information quickly and efficiently. This project aims to make the process of identifying gluten-containing products as efficient as possible and to smoothly incorporate technology into people's daily lives.

Conclusion

One of the most innovative software platforms created to speed up the process of identifying products that contain gluten is the Gluten Detection System. This paper provides project teams and new members with a thorough guide that explains the system's design and fundamental concepts. The feature showcases the smooth integration of technology, enabling users to quickly detect gluten in food items. Our application stands out in the market due to its unique capability allowing users to access gluten information about foods they wish to consume, regardless of their location, eliminating the need to be physically present in a store.

The system's fundamental components, a strong infrastructure, and an easy-to-use user interface, guarantee quick and effective information retrieval. User-centric filtering highlights the system's dedication to efficiency and usability by delivering tailored information quickly.

The main objective of the initiative is to make it easier for consumers to identify products that contain gluten while also incorporating technology into their daily life. Its unwavering emphasis on efficiency highlights the attempt to make technology a seamless and necessary component of people's daily lives.