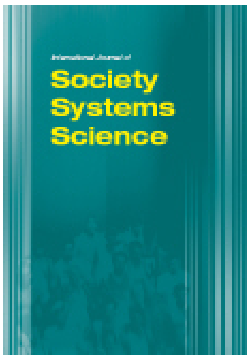


International Journal of Society Systems Science

ISSN online: 1756-252X - ISSN print: 1756-2511
https://www.inderscience.com/ijsss

Maverick: a smart mobile application-based automated system to combat food insecurity and ensure efficient monitoring for the school mid day meal feeding in developing countries

Md. Mehedi Hassan, Mahfuzulhoq Chowdhury

DOI: 10.1504/IJSSS.2023.10055785

Article History:

Received:	22 September 2022
Last revised:	23 November 2022
Accepted:	13 February 2023
Published online:	08 August 2023

Maverick: a smart mobile application-based automated system to combat food insecurity and ensure efficient monitoring for the school mid day meal feeding in developing countries

Md. Mehedi Hassan and
Mahfuzulhoq Chowdhury*

Computer Science and Engineering Department,
Chittagong University of Engineering and Technology,
Chittagong-4349, Bangladesh
Email: mehedirony22@gmail.com
Email: mahfuzulhoq.cse05@gmail.com
*Corresponding author

Abstract: To eradicate malnutrition problem in rural areas of developing countries like Bangladesh, the Government of Bangladesh started school feeding programme as a pilot project. But the current operational structure of the programme does not provide varying nutritious foods and has no reliable monitoring system. To eradicate the problems, this paper proposes a complete solution residing in a mobile application for school feeding programme. This paper provides a meal generation system based on remaining budget, attendance on each day, nutrition, and students' preference. After serving the selected meal, the meal serving process is monitored by first recognising students who were present before serving the meal through face recognition and then recognising if the selected meal was served through food recognition. The evaluation result verifies that our food recognition model achieves up to 98% precision and almost 80% of users are satisfied with our android application features for the school feeding programme.

Keywords: school mid day meal program; food menu planning; automation; monitoring; machine learning; mobile application.

Reference to this paper should be made as follows: Hassan, M.M. and Chowdhury, M. (2023) 'Maverick: a smart mobile application-based automated system to combat food insecurity and ensure efficient monitoring for the school mid day meal feeding in developing countries', *Int. J. Society Systems Science*, Vol. 14, No. 3, pp.222–246.

Biographical notes: Md. Mehedi Hassan received his BSc in Computer Science and Engineering from the Department of CSE, Chittagong University of Engineering and Technology in 2022. His research interests include machine learning and mobile application development.

Mahfuzulhoq Chowdhury received his PhD in Telecommunications from the University of Quebec, INRS, Montreal, Canada. He is a faculty member at the Department of Computer Science and Engineering, Chittagong University of Engineering and Technology since 2010. His research interests are related to machine learning, optimisation, mobile application development, game

theory, resource allocation in future generation networks, cloud computing, Society 5.0, IoT, age of information, and blockchain. He has published several research papers at highly cited IEEE journals, transactions, magazines, and conference proceedings as well as chapters of books.

1 Introduction

Malnutrition is a condition that is commonly seen in a person when the right amount of energy or nutrition intake is not ensured in a persons' daily food intake (World Health Organization, 2021). It is commonly seen in people, especially children, from poor and developing countries due to their lower living standards and lack of access to nutritious foods. Children in developing countries like Bangladesh suffer from persistently high levels of anaemia and chronic malnutrition, as well as persistently high rates of acute malnutrition. These problems have an impact on the health of schoolchildren, their ability to focus in class, and their potential for future productivity as adults. Since 2001 with an ambition to reduce malnutrition, the World Food Programme (WFP) has been feeding children in Bangladesh in collaboration with the Government of Bangladesh (Directorate of Primary Education, 2020). The Directorate of Primary Education (DPE) of the Government of Bangladesh has been taking the initiative to give khichuri (i.e., food items with lentils and rice) for four days and high-energy biscuits provided by WFP as a mid-day meal to students (i.e., primary and pre-primary schools) aged 3–12 to ensure 30% of the energy required per day aiming at enhancing attendance and reducing dropouts (Directorate of Primary Education, 2020). However, the honourable prime minister of Bangladesh raised objections to this project at a recent Executive Committee of the National Economic Council (ECNEC) meeting due to her worries regarding the project's framework, stating that preparing khichuri almost every day at school could impede students' education and encouraged for development of newer formats (bdnews, 2021). The prime minister also suggested that the students' regular meals should include some diversity.

In addition, India has been providing several hot, prepared meals in primary and high schools on all weekdays since 2004 under the mid-day meal scheme (Government of India, 2021). Mid-day meals at school are not only an encouragement for parents to send their children to school, but they also have a great impact on student's physical and mental health and education. This could be a multi-beneficial investment in human capital and the local economy. In our work, we have proposed a more structured solution to school feeding programme (SFP) in Bangladesh. The Government of Bangladesh's SFP, a national effort that is still a pilot operation, requires routine auditing and monitoring. As it is carried out by NGOs monthly, the existing monitoring method has significant shortcomings. There is a lot of potential for interruptions like food waste and illegitimate supplies. Further, for proper functioning the school feeding or mid-day meal scheme requires tackling several challenging research issues like proper food menu generation by taking budget and presenting student count, price of foods, the nutritional value of foods, students, authority and teachers interaction for monitoring, food budget availability, cooking issue, mismanagement, students awareness about nutritious food, students and guardians feedback about the SFP, budget collection, and corruption handling, among others (Directorate of Primary Education, 2020).

At present, a notable amount of literature works exist in the area of mid-day meal schemes or food menu generation for users. However, literature works on meal menu generation mainly consider limited food items rather than many. For example, the works in Haque et al. (2013) investigate the school feeding schemes outcome by taking only one or two food item-based menu plans. Similarly, the works in Directorate of Primary Education (2020) investigate a meal plan generation by taking four foods into account. Generally, it is really difficult to get proper nutrition amount by taking a few fixed food items. Also, based on the availability, a variety of food items for food menu selection would be appreciated by both students and guardians. Marrero et al. (2019) investigate food menu generation plans by taking cost and nutrition amount into account. Whereas, the works in Elseweiler et al. (2015) developed a balanced food menu plan by taking users' choices and nutrition into account. The works in Aguilar-Loja et al. (2022) emphasise on highest nutritional value-based food menu generation via a well-known machine learning technique. They did not investigate proper monitoring or multiple factor-based food menu generations.

Importantly, for proper meal menu plan generation scheme development for SFP several other issues need to be checked out like total budget, food current cost, available food items, and present student count number along with students' preference and calorie value. Another important research question is how to ensure proper monitoring of SFPs by using available technologies. To answer that question in Jayakumar et al. (2020) the authors proposed a face detection-based mid-day meal monitoring scheme. However, to ensure corruption-free monitoring of the mid-day meal scheme only face detection of students is not enough both real-time serving food item detection and present students face detection via attendance feature are also necessary. The works in Samal et al. (2019) emphasise only students' attendance system for the mid-day meal scheme. They utilised manual attendance, camera-based image capturing, students localisation value, and cloud processing service for student attendance verification. However, they did not investigate some important mid-day meal scheme research issues like multi-factor-based food menu generation, students' real-time food serving process monitoring, students feedback, interaction with students and teachers, per day menu cost and budget calculation, corruption-free evidence reporting to authority, among others.

From the previous work discussion, it is clear that the existing works did not investigate any suitable automated system for the SFP for developing countries like Bangladesh to tackle shortcomings like food insecurity and corruption monitoring. To eradicate different issues associated with the mid-day meal scheme, the fundamental research problem that this paper tries to solve is the development of an automated system that provides a proper meal menu plan by taking multiple factors into account such as different food items, food items cost and availability, budget per day and associated cost, students number count via mobile application-based attendance process, real-time food detection during food serving time to students, evidence collection process and authority reporting process, students and guardians feedback process, teachers authentication process, among others. To do so, this paper develops a smart mobile application that includes some interesting features like automated meal menu plan generation by taking cost, budget, present student number, student preference into account, students attendance taking process, real-time face and food detection for attendance and food distribution proof, monitoring, evidence attachment to authority for corruption avoidance, among others.

Above all, to improve the huge mismanagement, food insecurity, monitoring, and corruption issues of existing mid-day meal feeding systems, this paper presents an android mobile application-based automated system for the SFP in developing countries like Bangladesh. The notable features and contributions of this work are outlined below:

1. In this paper, a complete android application is built for SFP in Bangladesh by suggesting meals, ensuring maximum transparency in monitoring and sending reports to officials. The meal menu generation examines several factors like per day budget, nutritional values of foods, present students count number, students' choice, multiple menus and their costs, among others.
2. The best possible meal is selected per day by using our application based on the number of present students each day, the current budget remaining, and students' preferences, maximising energy intake. Starting off the application, we have counted the number of students present each day through face recognition using the state-of-the-art CNN face model (MobileFaceNet) (Chen et al., 2018). This application can ensure a meals preparation each day by taking students' attendance through recognising each student's face. The real-time student attendance count via a mobile application with authorised teachers is another notable contribution.
3. This paper also provides a mid-day meal monitoring and verification process. After serving a meal, the application verifies if each student was served the intended meal for the corresponding day by recognising the students' faces and detecting served foods. The foods are detected using EfficientDet (Tan et al., 2019), an object detection model that uses EfficientNet (Tan et al., 2020) as a backbone.
4. This proposed mobile application-based automated system offers a feature for corruption handling related to the mid-day meal scheme. After successful verification, the reports are sent to officials daily. The daily reports consists of the number of present students, served a meal each day, the cost of meal each day and the list of students who got a meal.
5. This work also can collect feedback from students/guardians for improving the programme and making options for community donations.

Next, Section 2 discusses the previous works. Section 3 gives an overview of the proposed automated system for the SFP. Section 4 discusses our achieved results in the application. Section 5 gives an overall conclusion of this work along with some future work suggestions.

2 Related works

This section includes a detailed discussion of existing literature works related to food menu generation, school feeding, face and food detection for monitoring, and reporting process associated with the mid-day meal scheme. In 2001, the WFP and the Government of Bangladesh began collaborating on the SFP (Directorate of Primary Education, 2020). A five-day meal plan with khichuri on 4 days and high-energy biscuits on each Thursday was devised by nutritionists (Directorate of Primary Education, 2020). However, a severe sickness outburst was documented in 2010 among children in northwest Bangladesh who were consuming high-energy biscuits as part of the SFP (Haque et al., 2013). This suggests the need for a balanced meal plan with variations taking into account nutritional needs, budget, preferences, etc. Based on these many parameters, an automated meal plan can provide suitable food menu solutions. Despite having a considerably simpler mathematical formulation, the menu planning problem (MPP) is an NP-Complete problem (Garey et al., 1979). By using precise deterministic methods, NP-complete problems are challenging to solve in polynomial time. According to Brazilian references, Marrero et al. (2019) created a meal plan for Brazilian schools to reduce cost and nutritional errors. Based on the user's taste preferences and calorie needs, Elsweiler et al. created a food recommendation system (Elseweiler et al., 2015). To gather personal data from users such as height, weight, age, and food preferences, they developed a website as a food portal. Then they recommended to the user a food menu that carried out their expectations. However, a significant flaw in their work was that they did not take into account food costs, which is a key parameter in devising a menu plan.

The mid-day meal scheme (MDMS) in Uttar Pradesh, India, was strengthened by Kadari et al. (2016) using management information system (MIS) and interactive voice response system (IVRS). They used IVRS, which automatically reached users through voicemail, took their response in digits 0 or 1 – meals served or not served – and updated the server through MIS, to guarantee that all students received meals each day. For the mid-day meal program in India, in Samal et al. (2019), the authors developed a two-factor system for verifying students' attendance each day. First, they registered students along with their images and trained a FaceNet (Schroff et al., 2015) model with these images, and stored them in the cloud. Then each day took both manual attendance and some landscape images of the students and sent them to the cloud to compare with the saved model. They used multi-task cascaded convolutional neural network (MTCNN) for detecting faces and then FaceNet (Schroff et al., 2015) extracts 128-bit face embedding from these faces. Finally, they classified faces from these embeddings using SVM. They did not use a real-time camera as input which creates options for corruption. In addition, their two-factor verification can be simplified in one step by detecting faces on the camera screen. The works in Samal et al. (2019), and Schroff et al. (2015) discussed only a single feature like food menu planning or face detection for attendance only rather than multiple features for automated school feeding mobile applications.

In Jayakumar et al. (2020), the authors implemented a system for monitoring mid-day meal programs in India. They used Viola et al.'s (2004) face detection algorithm. There are some limitations in their work, they only counted several unique faces and unique foods in the camera frames. They did not classify which students were served which food. This opens up paths for corruption. Another limitation was they used computers for overall monitoring, which is not very much likely for schools in the village. In Jiang et al. (2020), the authors created a deep CNN model for food classification using faster R-CNN on food images to construct the region of interest (ROI) and VGG16 (Simonyan and Zisserman, 2015) for extracting the feature map to classify each food. They created a dietary evaluation by assessing the calorie, carbohydrate, and protein values of the food's constituents based on the identified results. In Kagaya et al. (2014), the authors developed a food detection and recognition system using a convolutional neural network with a dataset of ten classes of food images. They showed a comparison between different existing food recognition techniques with an accuracy of 52–60% and their CNN model outperformed them with an accuracy of 73.70% on average. They also worked on food detection of whether an image is a portion of food or a non-food image with 93.8% accuracy. Different important school feeding mobile application features like students attendance count, monitoring food distribution, and multi-criteria-based menu planning were out of the aforementioned work studies (i.e., Jiang et al., 2020; Simonyan and Zisserman, 2015; Kagaya et al., 2014).

In Bettadapura et al. (2015), the authors developed a method for automating food journaling from photographs of foods taken in restaurants serving five different cuisines by inferring geolocations and detecting foods in the photos. Using hierarchical segmentation, they retrieved areas where foods are predicted to exist. They trained and evaluated using the SMO-MKL SVM classifier. With past location, the average accuracy was 63.33% because it narrows the search field; without prior location, accuracy drops to 15.67%. Bai et al. (2021) utilised lidar as well as machine vision technology for a mobile robot-based meal delivery system. In Segredo et al. (2020), the authors presented an automatic and balanced menu plan application using a genetic algorithm for schools. Aguilar-Loja et al. (2022) used a decision tree classifier to recommend the best nutritional menu plans. However, important issues like proper corruption monitoring, students' real-time attendance, participation, and real-time feedback regarding the meal serving process were out of their investigations (i.e., Segredo et al., 2020; Aguilar-Loja et al., 2022; Bettadapura et al., 2015; Bai et al., 2021) . To offer not only traceability but also accountability, Nukapeyi et al. (2022) detailed a blockchain-based system for a mid-day meal scheme. To ensure workers' hygiene along with safety issues, Kuntagod et al. (2021) discussed a prototype using CCTV cameras, IoT, and artificial intelligence techniques. They only emphasise monitoring the cooking areas rather than other parts of the system. However, both works (Nukapeyi et al., 2022; Kuntagod et al., 2021) did not investigate several features like attendance menu, budget and nutrition-based automated food menu generation, real-time food detection, and student identification.

From the aforementioned related works discussion, it is clear that most of the works considered only one type of objective like only food menu planning rather than many other school feeding application features. Importantly, several schools feeding mobile application features like real-time attendance features, budget, nutrition, and student preference-based menu selection, face and food detection, feeding process and corruption monitoring were out of their proposal. However, different from the existing research works, this paper develops an android application-based automated system for the SFP in developing countries by taking both the food menu planning feature, students attendance feature, food, and student detection feature, and proper monitoring feature, donation into account.

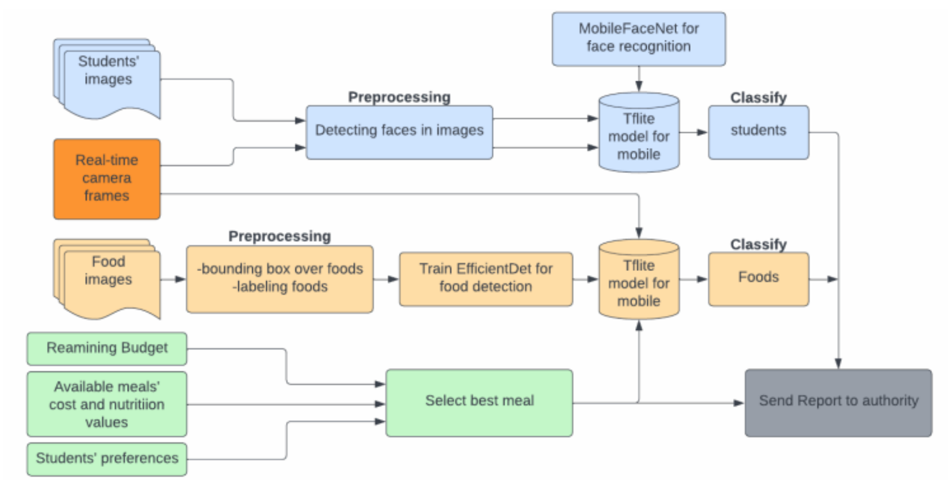
3 Proposed system model

This section details the overview of the proposed model (named as maverick). Figure 1 shows the proposed maverick system's abstract overview. For face recognition at first, the system detects the region of a face in each image stored in the phone and using MobileFaceNet (Chen et al., 2018) compares them with the extracted face from the real-time camera and classifies each face from camera frames. For food recognition, we collected several images of some easy-to-prepare foods and annotated each of them with bounding boxes and labels. Then we trained EfficientDet (Tan et al., 2020) model with these labelled images. Both of the models were converted into the TensorFlow lite (tflite) model to be compatible with mobile phones. Now the application can recognise students and foods from the real-time camera frames. Then we selected the best meal from a set of available meals considering budget, nutrition values, and students' preferences. For monitoring, the application verifies if each of the present students on each day was served the selected meal. Finally, the application sends each day's report to the authority. Our proposed system framework can be classified into two major phases:

1 training two CNN models for face recognition and food recognition

2 developing an android application that interacts with these models along with several other features.

Figure 1 Overview of the proposed maverick system (see online version for colours)



3.1 Overview of training the face and food recognition models

Our detailed model regarding face recognition and food recognition is illustrated in Figure 2. For face recognition we used MobileFaceNet (Chen et al., 2018) and converted it into TensorFlow-lite to be compatible with mobile. The model extracts a 128-d vector for each face. Thus we can compare our stored faces of the students with the faces extracted from the real-time camera. We then classified the face using cosine similarity between these two embeddings. For food recognition firstly we created a dataset of food images consisting of eight easy-to-prepare foods. We then annotated each of the images with a bounding box and label each of the foods using LabelImg. Then we used this dataset to train EfficientDet-lite2 (Tan et al., 2020) on our dataset. Finally, we converted the model into a TensorFlow-lite version that can recognise the foods that it is trained on.

Figure 2 Steps of training our face recognition and food recognition models (see online version for colours)

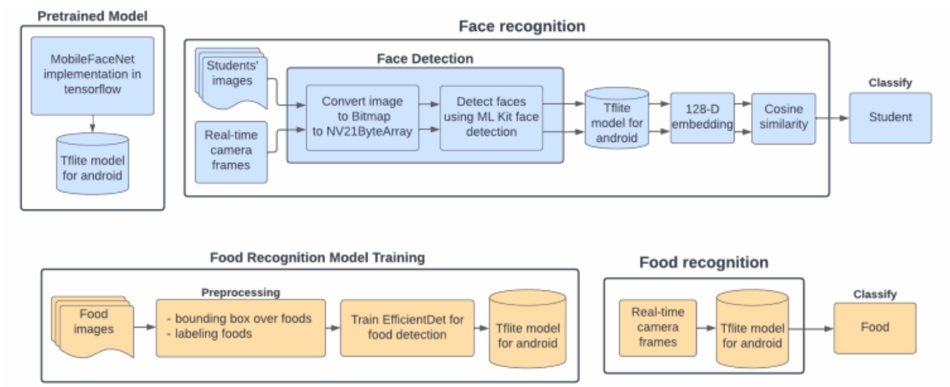
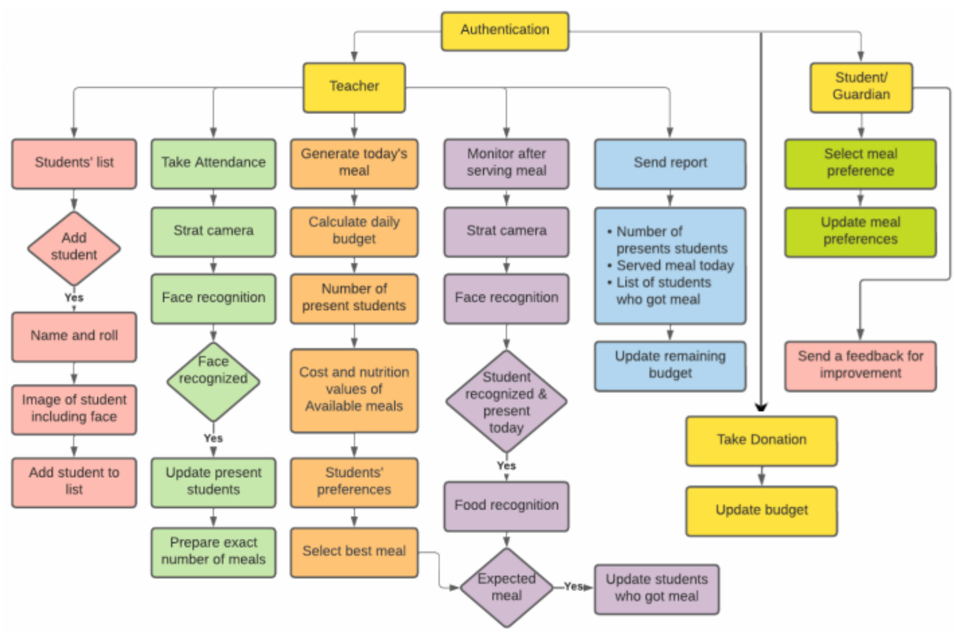


Figure 3 Overview of our proposed Maverick application (see online version for colours)



3.2 *Overview of the application*

Figure 3 shows our Android application (Maverick) for SFP comprises several features that facilitate interaction between teachers and students of a class. While authenticating, a user can authenticate as a teacher or student, each of which has different use cases. A teacher can see the list of students in his class and add students. Each of the students will have a roll number, name, and some images that include their faces, based on which they will be recognised later. The teacher can take attendance using face recognition Figure 2 which counts the number of present students. Each day the best meal is selected from a set of available meals based on the remaining budget, the number of present students, and students' preference for foods over the week. After serving the food the teacher just needs to open the camera and the application will recognise each student along with their served food and update the list of students who received meals on that day. Finally, the teacher should send the daily report to the authority. A student can submit their meal preferences and guardians can give feedback too for improving the program. The application also features a donation profile that can be accessed irrespective of authentication.

3.2.1 Data collection

As we had to work with two types of data – face image and food image, our dataset preparation process for these two use cases has different procedures.

3.2.2 Students' data collection

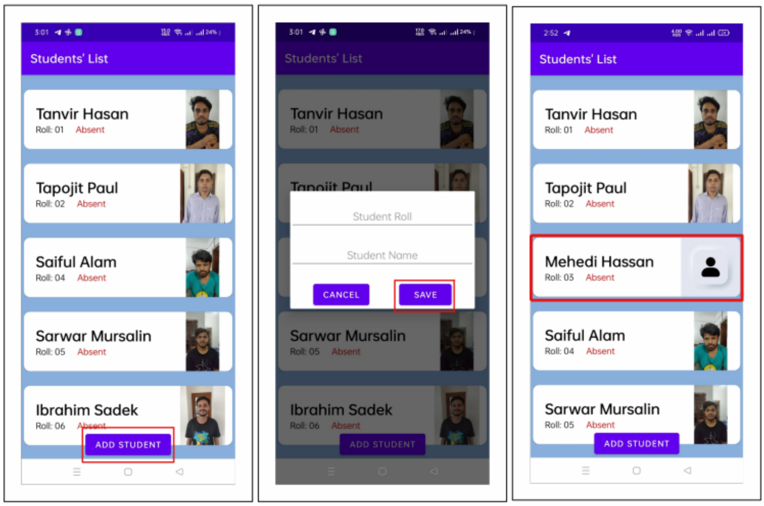
The number of students in a class is not always fixed, it varies from month to month. Taking this into consideration we decided to make adding students' information easier for teachers and made procedures such that our user's face recognition model is trained on the students' faces stored on the phone. We intentionally omitted the scope for uploading and downloading face images from the cloud to make the system more teacher-friendly for schools in villages making the use of the internet as less as possible. In the case of sharing the face images, teachers just need to share the face image folder on the receiver's phone. Now for taking pictures of students we followed the following procedure.

A teacher will have access to check and update the list of students. Figure 4 shows the steps of adding a student. Clicking on 'add student' in Figure 4(a) will create a dialogue like Figure 4(b) asking to enter the student's name and roll. Entering expected info and saving it will create a new instance for a student shown in Figure 4(c) A teacher can add images for any student by clicking on the student's instance in Figure 4(c). Student details can be seen in Figure 5(a). A teacher should add 3–4 images for each student. Clicking on 'add a face' in Figure 5(a) will create a camera preview for taking a picture. Capturing an image in Figure 5(b) will store the image in the respective directory in phone storage. The captured image for the respective student can be seen in Figure 5(c). More faces can be added by clicking on 'add a face'. The stored images will be used later for recognising the students using our face recognition model.

3.2.3 Food data collection

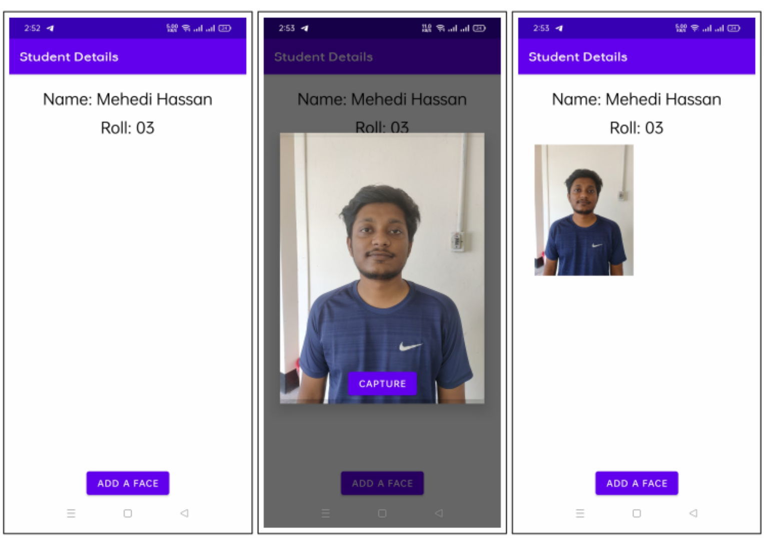
We need to focus on both food menu preparation and food image dataset preparation. In our work, we have selected 8 easy-to-prepare foods that will be later combined to select meals each day. The energy and cost of our selected foods considering serving size for a single person are shown in Figure 6(a). The nutrition values are taken from different sources and costs are calculated considering the current market price. We have collected nutrition data from the Bangladesh Government Primary Education Directorate (DPE) website and Nutritionix website. Due to changes in market price, these costs can vary over time. The energy and cost of our selected meals considering serving size for a single person are shown in Figure 6(b). These values are calculated from the values in Figure 6(a). We have tried to select meals that will be both healthy and preferable to students.

Figure 4 Overview of our proposed application (see online version for colours)



(a) Students' list (b) Add student's roll and name (c) New student added

Figure 5 Process of adding student's image (see online version for colours)



(a) Student details (b) Capture image (c) New picture added

We have created a dataset of 631 images of our eight selected foods. For each of the food items, we collected images from Google images and there are several of our own captured images. We then annotated each of the foods in the next phase which will be explained later.

Figure 6 Food items and nutrition values

No	Food items	Energy (kcal)	Cost (Taka)
1	Khichuri	577	34
2	Mixed Vegetable	188	18
3	Milk	122	20
4	Egg	92	9
5	Paratha	258	10
6	Bread	231	7
7	Banana	105	7
8	High Energy Biscuit (Provided by WFP)	337	10

(a) Selected food items

No	Meals	Energy (kcal)	Cost (Taka)
1	Khichuri	577	34
2	Khichuri and Egg	669	43
3	Milk and Biscuit	459	30
4	Paratha and Mixed Vegetable	446	28
5	Milk and Bread	353	27
6	Paratha and Egg	350	19
7	Banana and Biscuit	442	17
8	Bread and Egg	323	16
9	Bread and Banana	336	14

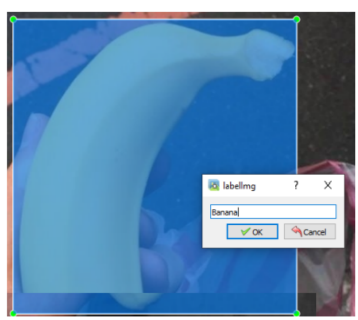
(b) Selected meals

3.3 Data preprocessing for face recognition and food images

For recognising a face we first need to detect a face in an image. Face detection is the process of detecting whether there is a face in an image. If there is a face, it extracts a bounding box over the face. A bounding box is a rectangle with four properties – left, top, right, and bottom. These detected faces can be used later for recognising a face. A bounding box over a face extracted by face detection is shown in Figure 7(a). Each of the student's images stored on the phone is compared by the face recognition model for recognising faces. For face detection, we have used ML kit face detection API. For the API the input image should be in NV21 format. So we first need to convert each of the students' images and frames from the real-time camera into NV21 format. Then we can pass an image for detecting faces. Finally, we get a bounding box in the image if there is a face in there.

Figure 7 Data pre-processing (see online version for colours)

(a) Face detection



(b) Food annotation

The food images need to be preprocessed before using them for training the food recognition model. In preprocessing step, we annotated each of our collected food images using LabelImg. We annotated the images with a bounding box and a label for each food in an image. These annotated images are then imported in Pascal VOC format for training. Pascal visual object classes (VOC) is a format that stores annotations for object detection datasets in an xml annotation file that consists of the label and bounding box for each annotated image. While training, the model then can recognise the foods along with a bounding box over the recognised food. Figure 7(b) shows an annotated example using LabelImg.

Figure 8 Summary of the food recognition model (see online version for colours)

```
[13] model.summary()

Model: ""
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 keras_layer (KerasLayer)    multiple                  5227280

 class_net/class-predict (Se  multiple                 9144
 parableConv2D)

 box_net/box-predict (Separa  multiple                 5076
 bleConv2D)

=================================================================
Total params: 5,241,500
Trainable params: 5,168,940
Non-trainable params: 72,560
_________________________________________________________________
```

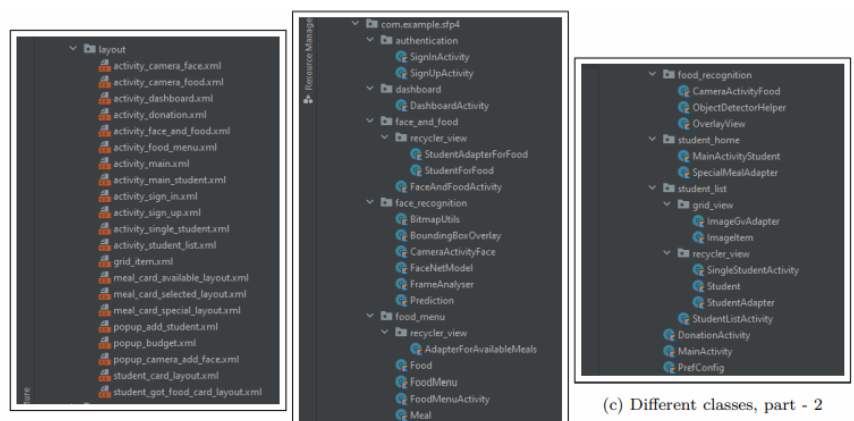
3.3.1 Training the face and food recognition models

For face recognition, we have used MobileFaceNet (Chen et al., 2018), a state-of-the-art face model that is trained on MS-Celeb-1M (Guo et al., 2016) face dataset. This 128-d embedding can be compared with other extracted embeddings to recognise a face. For our case, we converted the MobileFaceNet model into a TensorFlow-lite model such that it can run on mobile to satisfy our cause, and recognise students' faces. The students' images are stored in the phone that is captured by the teacher. When the camera is started, the application fetches the students' stored images and gets the detected faces following the steps explained in the previous section. Then the frames from the camera are fetched and following the same procedure it gets the detected faces. For each frame, we consider the extracted face from the camera as our subject and compare this subject with our previously extracted students' faces. For comparing two faces we used cosine similarity. Cosine similarity is a similarity measure between two vectors that measures if two vectors point in the same direction (Han et al., 2012). Let for our face recognition case, we tend to compare the extracted 128-d face embedding of our subject from camera frame, s and face embedding from the actual stored face of one student, a. Then the similarity equation for n = 128 will be calculated or cos (s, a) based on Han et al. (2012). For each student, we calculate the cosine similarity and finally select the student that yields the maximum similarity as the cosine of minimum angle results in maximum value. Selecting the maximum cosine similarity or minimum angle between two vectors maximises the probability of two faces being of the same student. Then we classify our subject from the frame as the recognised student. All of these recognition training on our created students dataset are run on the phone. The working procedure explained here is illustrated in Figure 2. For food recognition firstly we created a dataset of food images. After preprocessing the dataset we import the Pascal VOC formatted data before training. Then we used this dataset to train EfficientDet-lite2 (Tan et al., 2020) on our dataset. We evaluated the trained model on our validation data. A summary of the model can be seen in Figure 8. For each food image, it predicts a class and a box. The total number of parameters is around 5.24 million where 5.16 million are trainable parameters and 72.56k are non-trainable parameters. Finally, we created a TensorFlow-lite version of the model that can detect any food on a mobile device using the camera in real time. The training procedure is shown below in Figure 2.

3.4 Result and analysis

Our major goal was to develop an android application that could incorporate several features that the SFP lacks and make improvements in the operational structure of the program. For each of the screens of the application, we made a layout and an activity. The layout files are written in XML which contains the design of the screen layout and the activity files are written in Kotlin which determines the interaction within the layouts and the interaction between two or more layouts. An overview of the needed layout files and classes/Kotlin files are shown in Figure 9. For storing and retrieving data we have used Firebase Firestore. For training our CNN models we created two types of data. One dataset is for face recognition and the other is for food recognition. For face recognition of students, firstly we collected a dataset of 40 students from a student hostel named Qudrat-E-Khuda Hall, CUET following the data collection procedure we explained in the previous subsection. For food recognition, we created a dataset of 631 images of eight food items. Several of these images were collected from Google images and several were captured using our phone. The description of the dataset can be seen in Figure 6.

Figure 9 Overview of the implementation in Android Studio (see online version for colours)



(a) Layout files

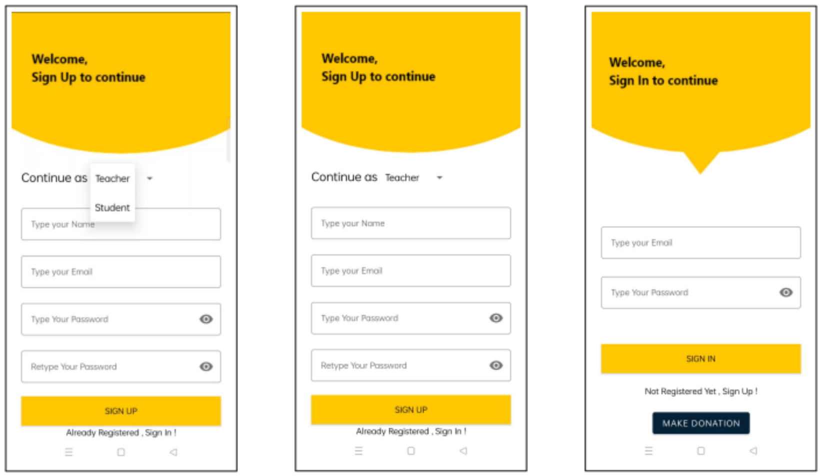
(b) Different classes, part - 1

(c) Different classes, part - 2

3.5 Android application description

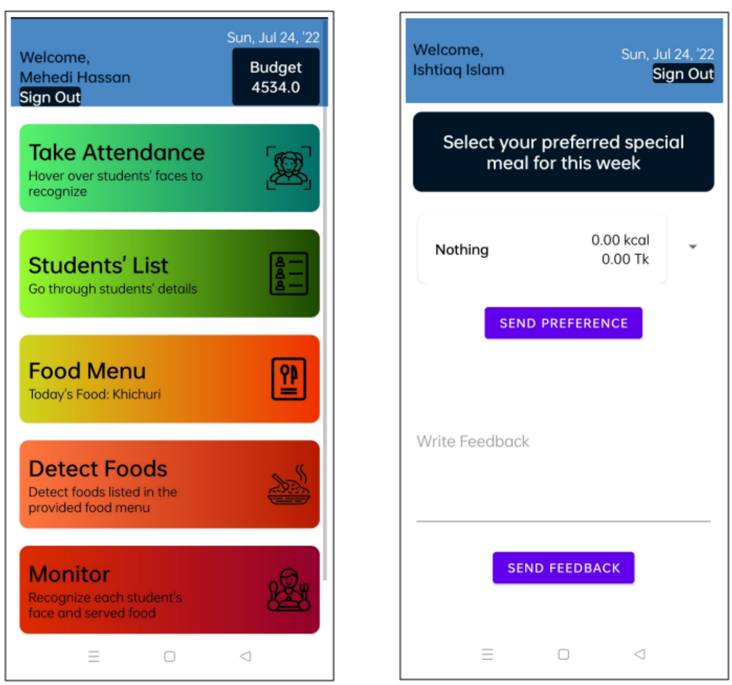
During sign-up, a user will have to choose a role, teacher or student, like in Figure 10(a). After selecting a role, the user needs to fill in the details required in Figure 10(b) to sign up. After successful registration, the user now can sign in to the system.

Figure 10 Authentication process (see online version for colours)



(a) Selecting role while registering (b) Required details for signing up (c) Sign in page

Figure 11 Home page (see online version for colours)



(a) Teacher's home page (b) Student's home page

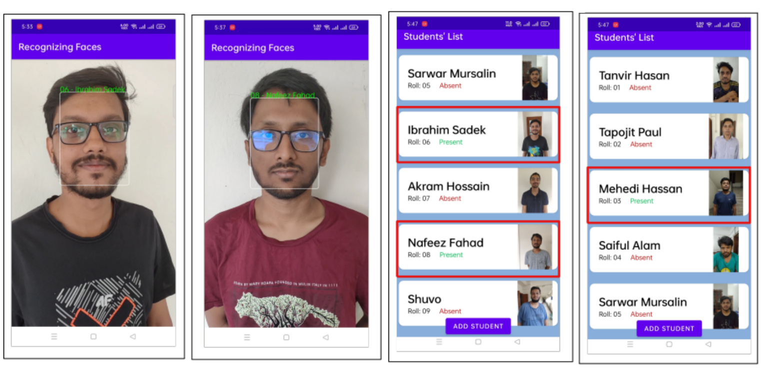
3.6 Home page

The system will automatically direct users to their respective home pages according to their role – teacher or student, only from their email. Home pages for teacher and student are shown in Figure 11(a). In the top left, the user's name will be fetched from Firestore and shown for the user's sign-in confirmation. A teacher has options to choose from a set of actions: taking attendance for confirming the number of meals to prepare, checking/updating students' lists, generating meal for each day, check if the system can recognise foods that it has been trained on, monitoring as in recognising each student along with their served meals and finally sending a report to the authority for each day. Similarly, like Figure 11(b), a student/guardian can select meals that they want each week as special meals and can send feedback about the program to the authority.

3.7 Taking attendance

After signing in as a teacher, with their respective emails, a teacher will have access to several features. Each day the application shows the current date in the top right of the screen. Starting every day, a teacher needs to take attendance of the students to ensure a number of meals are prepared each day. A teacher's home page can be seen in Figure 11. From there he/she needs to click on 'take attendance'. Then the application will start the camera for taking attendance. The teacher just needs to hold the camera over the students' faces as shown in Figure 12. The application will recognise faces following the face recognition procedure we explained before in the methodology. Multiple students can be recognised at once. Thus taking attendance using facial recognition is much faster than traditional approaches.

Figure 12 Taking attendance (see online version for colours)

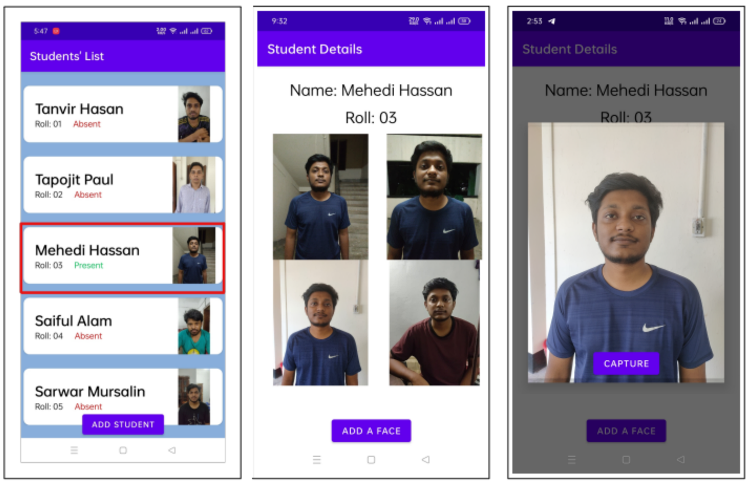


(a) Recognizing students (b) Recognizing students (c) Attendance updated (d) Attendance updated

3.8 Students' information

Figure 13 shows the information of the students in a class. From the home page in Figure 11(a) a teacher can check and add students' details by entering them into 'students' list'. Figure 13(a) shows the list of students in the class. Each of the students' instances consists of their name, roll, current presence state, and an image for identifying them correctly. Clicking on any of these instances will direct to the 'student's details' page in Figure 13(b). Student details contain the name, roll, and some images of the student that are used for recognising them using face recognition. More images can be added by clicking on the 'add a face' button which will create a camera preview for taking pictures shown in Figure 13(c). Clicking on 'capture' will take a picture and add the image to 'student details' page in Figure 13(b).

Figure 13 Student list and details (see online version for colours)



(a) Students' list (b) Student Details (c) Adding a new face

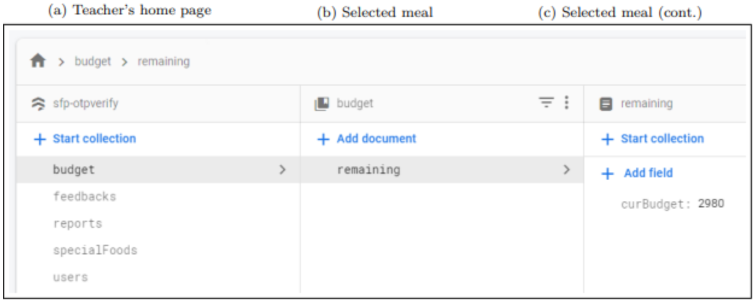
3.9 *Food menu generation*

After taking attendance, the application automatically selects the best meal for each day from our chosen meals in Figure 6 considering the remaining budget, number of present students and energy of the meals. The meal generation algorithm is explained in the previous section's meal generation section. For each day, firstly the system ensures that the number of present students is non-zero and the current day is not a weekend (Friday). Otherwise, it does not generate any meals. If requirements are met, the system selects the best meal for that day using our meal generation algorithm. Figure 14 shows meal selection for different cases. In the top right of the teacher's home page in Figure 14(a), the current remaining budget is fetched from the Firestore database shown in Figure 14(d). Then based on the remaining budget, the system chooses the best possible meal within the budget. Examples of meal selection based on budget and number of present students can be seen in Figures 14(b) and 14(c).

Figure 15(a) shows the current week's preferences of the students and Figure 15(b) shows that 'khichuri and egg' selected as a special meal on Thursday. The system stores new preferences in distinct collections once a new week starts. Thus the authority can check all of the previous weeks' preferences too from Firestore.

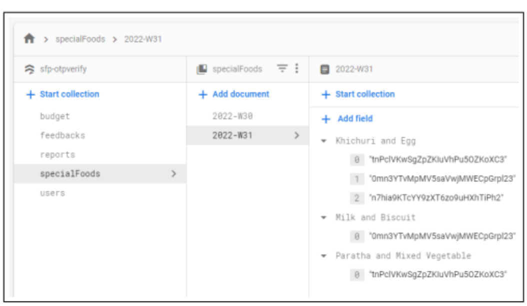
Figure 14 Food menu generation (see online version for colours)

(a) Teacher's home page (b) Selected meal (c) Selected meal (cont.)

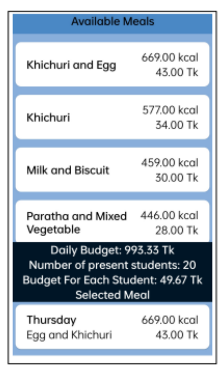


(d) Current budget stored in Firestore

Figure 15 Special meal selection (see online version for colours)



(a) Students' meal preferences stored in Firstore

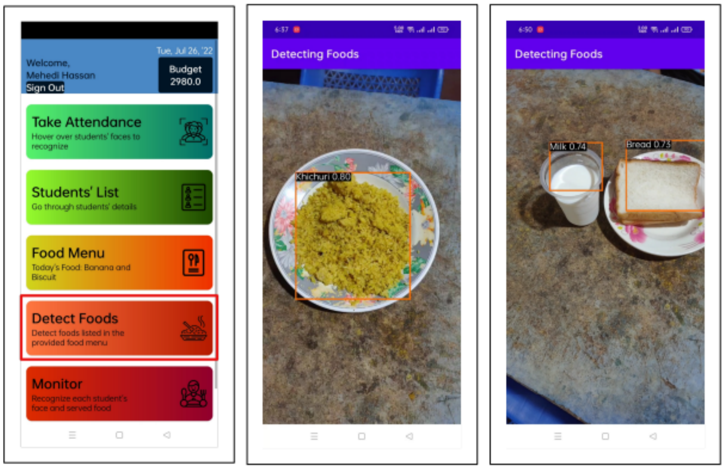


(b) Selected special meal on Thursday

3.10 *Detecting foods*

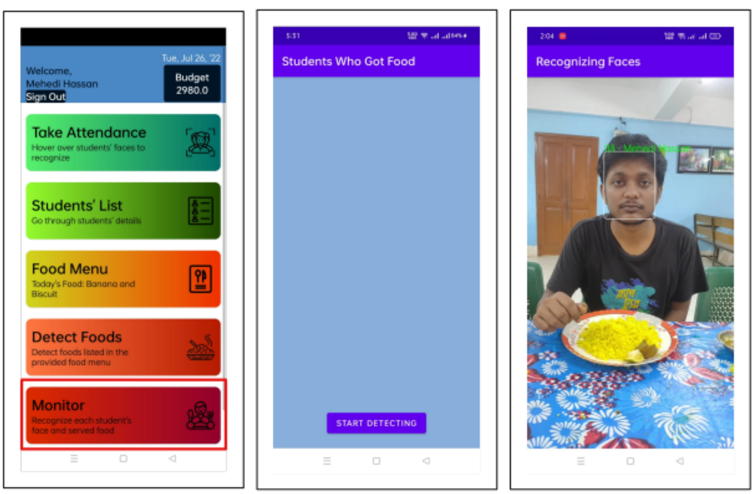
Before serving the meals, the teacher can check whether the camera can detect the foods prepared that day. Figures in Figure 16 show the process of food detection. From the teacher's home page shown in Figure 16(a), a teacher can select 'detect foods' which will open the camera for detecting foods. Figures 16(b) and 16(c) show examples of food detection such as khichuri, milk and bread.

Figure 16 Detecting foods (see online version for colours)



(a) Teacher's home page (b) Detected Khichuri (c) Detected Milk and Bread

Figure 17 Monitoring process (see online version for colours)

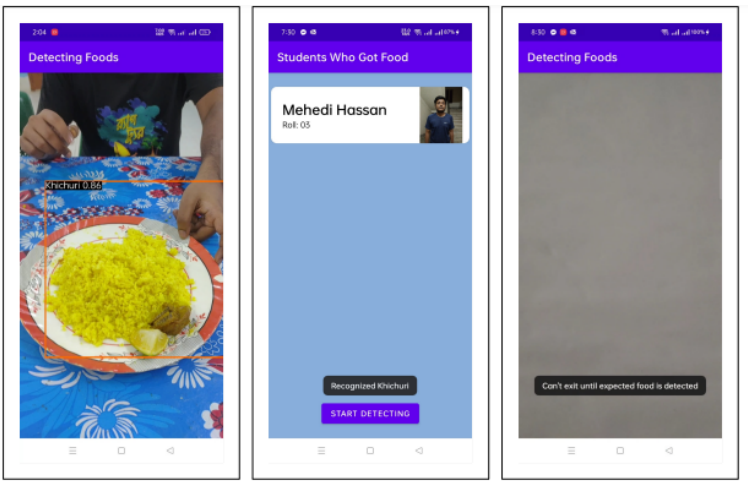


(a) Teacher's home page (b) Empty list of served students (c) Recognizing face

3.11 Monitoring

After preparing the expected meal automatically selected before each day, the meals will be distributed among the students. The monitoring process is described in Figures 17 and 18. After serving meals to the students, the teacher needs to click on 'monitor' from the home page shown in Figure 17(a). It will show the current list of students whom the system successfully recognised as students who got food shown in Figure 17(b). As no students have been detected till now, it shows no student. Clicking on ‘start detecting’ at the bottom of the screen will start the camera for monitoring. At first, the system will recognise a student shown in Figure 17(c). If the student is recognised and he/she was present while taking attendance at the start of the day, it will start detecting meals shown in Figure 18(a). If the recognised meal is the expected meal on that day, the system will successfully add the recognised student to the served student list shown in Figure 18(b). Following the same process, the teacher needs to add the served students to this list. To avoid corruption, a teacher cannot exit the screen for detecting foods until all of the food items in the expected meal are not detected by the camera [see Figure 18(c)].

Figure 18 Monitoring process (continued) (see online version for colours)



(a) Recognizing foods (b) Updated list of served students (c) Disabled exit access

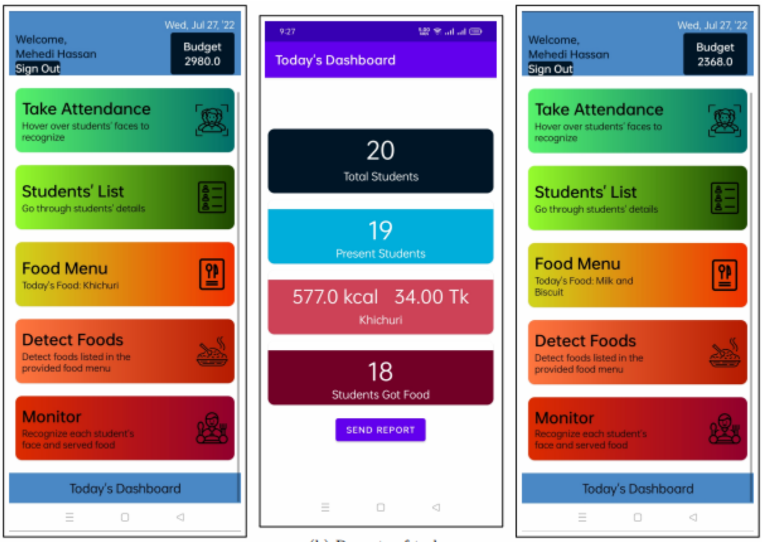
3.12 *Dashboard*

From the bottom of the teacher’s home page in Figure 19(a), he/she can check each day’s report and send the report to the authority. Figure 19(b) shows the current day’s report with the total number of students, the number of present students, the selected meal today, its energy, cost and finally the number of students who were served the selected meal on that day. The budget difference between before and after sending the report can be seen in Figure 19(a) and 19(c). The stored report in Firestore can be seen in Figure 20.

3.13 *Students’ meal preferences*

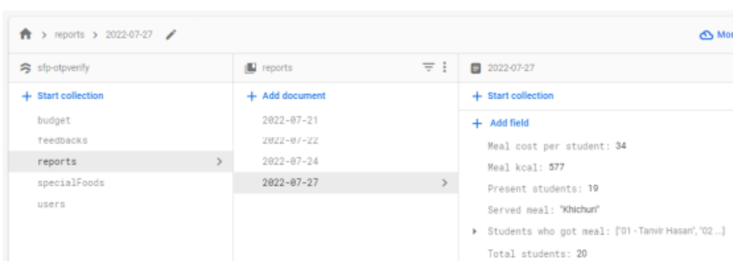
After signing in as a ‘student’, a student will have options to select meals as their preferred special meal for the current week. Initially ‘nothing’ is selected as a special meal. For selecting a special meal, the student needs to click on the dropdown marked with a red box in Figure 20(a). The dropdown includes all of the currently available meals along with their energy and cost. Selecting any of them will store the selections for that student [see Figure 20(b)]. A student can select multiple meals as their preference. Confirmation for each selection is shown in Figure 20(c). Finally, the student will have to send his/her preference to the authority by clicking ‘send preference’ in Figure 20(c).

Figure 19 Dashboard of each day (see online version for colours)

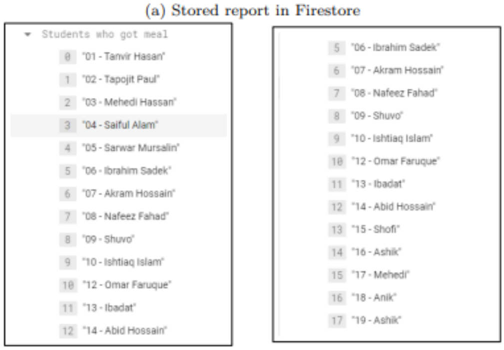


(a) Teacher’s home page (b) Reports of today (c) Updated budget

Figure 20 Saved report for each day (see online version for colours)



(a) Stored report in Firestore

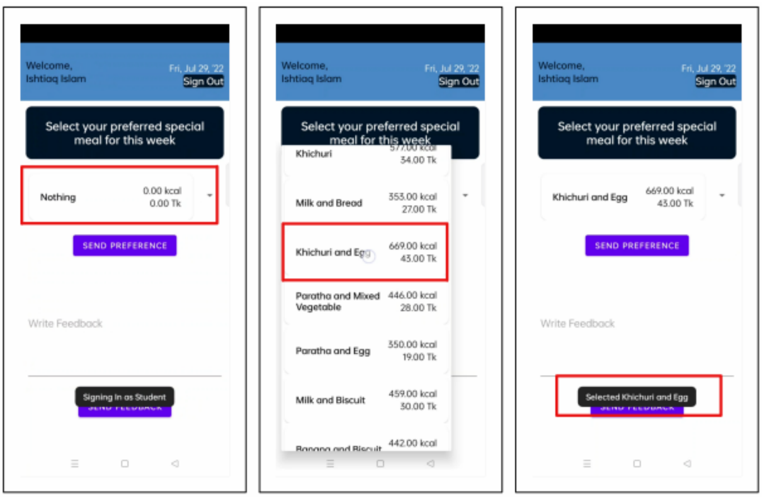


(b) List of served students (c) List of served students (cont.)

3.14 Sending feedback and donation

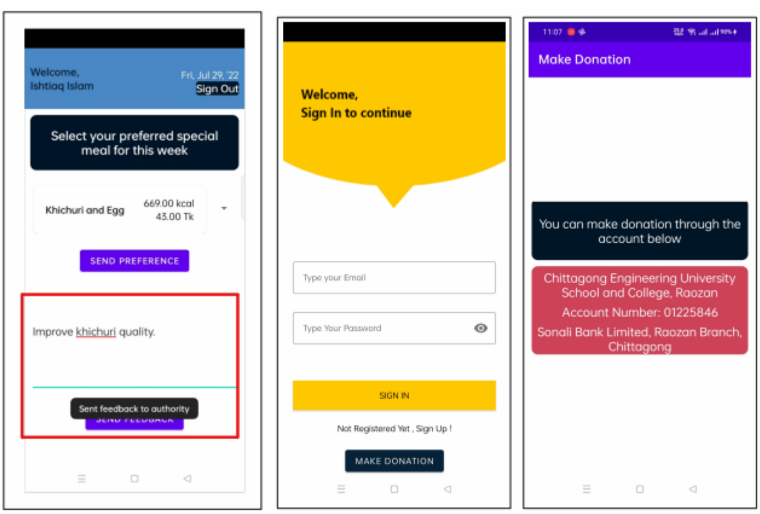
Students or their guardians can send their feedback for improving the overall program. From this feedback, the program authority can take action to improve the overall functional structure of the program. An example of sending feedback is shown in Figure 21(a). The process of donating is shown in Figures 21(b) and 21(c). From the sign-in page, clicking on ‘make donation’ at the bottom will open a donation profile for the respective school authority.

Figure 21 Process of sending meal preference (see online version for colours)



(a) Student’s home page (b) Selection of preference (c) Sending preference

Figure 22 Sending feedback and donation (see online version for colours)



(a) Sending Feedback (b) Sign in page (c) Donation page

3.15 Novelty of this research with efficacy

The novelty of this work is it provides several useful mobile application features at the same time for the automated SFP. Similar existing works like Elseweiler et al. (2015), Marrero et al. (2019), Kadari et al. (2016), Samal et al. (2019) and Jayakumar et al. (2020) investigated only a single feature (i.e., either menu planning or face detection for monitoring) for the SFP. To provide a smart mobile application-based automated system for school-going students and teachers to combat food insecurity (i.e., by providing nutritious foods) and ensure efficient monitoring of the mid-day meal scheme for developing countries. Differ from the existing works, the proposed application provides several important features at the same time like per-day food menu generation, student attendance support, per-day meal cost calculation and feeding monitoring process, among others. Differ from the existing works, our work provides cost and nutrition values of daily available meals along with students' preferences, attendance taking procedure, multi-criteria-based best meal selection for each day, face detection of students, food detection for monitoring each day activities, the interaction between teachers and students via a mobile application, authentication of teacher and students, list of present students generation via attendance, per day cost and meal generation, daily reporting to authority regarding per day served food and cost with evidence (to avoid corruption), students and guardians feedback regarding meal feeding system, and donation collection, among others. The main efficacy of the proposed system is it would be very much helpful (in terms of time and cost) for the automated school mid-day meal distribution service along with proper monitoring and per-day attendance count. The user review results in Figure 24 shows clearly the necessity of the proposed scheme (i.e., almost the majority of the users rate it good) in terms of meal menu generation, food generation and distribution monitoring, sending the report, and attendance, among others.

4 Evaluation of performance

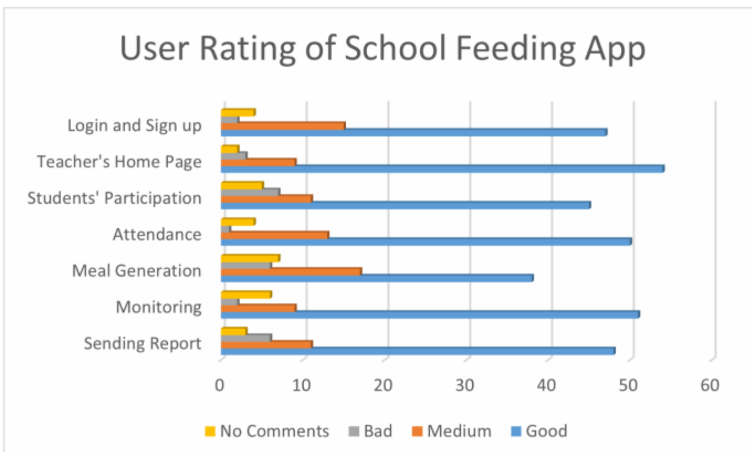
For evaluating the performance of the models this paper used the average precision value of food recognition models. For a prediction class, precision is the ratio of the number of successful positive predictions and the number of positive predictions by a model. Precision = TP/TP + FP. How well a model predicts is determined by precision. The average precision of our trained food recognition model is given in Figure 23. In object detection, intersection over union (IoU) is a similarity measure between two bounding boxes over an object, the area of intersection over the area of union. Here average precision is calculated over IoU thresholds from 0.50 to 0.95 with an interval of 0.05. AP50 is calculated at the IoU threshold of 0.50. AP75 is calculated at the IoU threshold of 0.75. The remaining APs are average precisions of our 8 food classes. From Figure 23, we can see that our proposed model can obtain up to 98% precision. The simulation model used for detecting foods and the monitoring process is detailed in Sections 3.2.1, 3.3, 3.9, and 3.10, respectively.

Further, we have surveyed how our proposed application performed in the context of SFP. Sixty-eight students participated in the survey. For the survey results, we have used both online and offline interview processes. The online survey with different application feature-related ratings and questions is conducted via Google survey form with questions. The offline survey was conducted via a face-to-face interview process (i.e., by asking different application features-related questions).

Figure 23 Average precision of our food recognition model

```
'AP': 0.83319813,
'AP50': 0.9854851,
'AP75': 0.9635225,
'AP_/Banana': 0.8402768,
'AP_/Biscuit': 0.9210585,
'AP_/Bread': 0.78995085,
'AP_/Egg': 0.7933494,
'AP_/Khichuri': 0.8555138,
'AP_/Milk': 0.7923091,
'AP_/Mixed Vegetable': 0.8553584,
'AP_/Paratha': 0.81776804,
```

Figure 24 User rating of school feeding application (see online version for colours)



We have taken their ratings on different features of the application on 4 scales: good, medium, bad, and no comments. The features we selected to take rating were: log in and sign up, teacher's home page, students' participation in the program, taking attendance, meal generation, monitoring and finally sending reports to the authority. The graphical illustration of the survey outputs is shown in Figure 24. The survey results depict that almost more than 80% of users are satisfied (i.e., the sum value of both medium and good rating) with different application features. The number of bad ratings is very low. Some reviewers also suggested during the interview that to improve the security of the proposed application features advanced security systems like blockchain are necessary. Finally, the figure notifies that the user rating of the school feeding application is very much satisfactory.

5 Conclusions

This paper presents and develops a smart mobile application-based automated system to combat food insecurity by providing a nutritious food menu and efficient meal-feeding process monitoring for schools in developing countries. Our proposed android mobile application tried to solve several existing problems related to the mid-day school meal feeding process by introducing new features such as meal suggestions based on cost, budget, nutrition, and students' choice, taking students' meal preferences, recognising students using facial recognition, attendance system, feeding process monitoring and real-time food detection, per day cost and meal feeding count reporting with evidence to avoid corruption, donation for school meal feeding, recognising students' served meals and many more. The detailed steps including face recognition for students' attendance and food recognition for per-day reporting were also discussed in this paper. The meal menu generation process investigates several crucial features per day students count, cost of menu and nutrition value, students' preferences, and students' budget, among others. This paper also performed user evaluation and more than 80% of users give satisfactory ratings in terms of the proposed application meal menu generation, monitoring, attendance, student participation, and report-sending features. The application connects several people in the process – teacher, student, and authority. Thus our system improves the overall transparency of the SFP. The proposed meal menu generation feature, monitoring, and reporting feature of our proposed mid-day meal application can be utilised in several institutions like orphanage hostels, hospitals, university hostels, senior citizen hostels, and prisons, among others. The future extension of our work will be adding more easy-to-prepare food items and increasing the size of the food dataset along with increasing high-security measures.

References

Aguilar-Loja, O. et al. (2022) 'A decision tree–based classifier to provide nutritional plans recommendations', *17th Iberian Conference on Information Systems and Technologies (CISTI)*, pp.1–6, DOI: 10.23919/CISTI54924.2022.9820144.

Bai, J. et al. (2021) 'Research on key technologies of meal delivery robot based on the fusion of lidar and machine vision', *7th International Conference on Big Data and Information Analytics (BigDIA)*, pp.236–242, DOI: 10.1109/BigDIA53151.2021.9619698.

bdnews (2021) *ECNEC Sends Back Primary School Meal Project for Review* [online] https://bdnews24.com/economy/2021/06/02/ecnec-sends-back-primary-school-meal-project-for-review (accessed 8 July 2021).

Bettadapura, V. et al. (2015) 'Leveraging context to support automated food recognition in restaurants', *Proceedings – 2015 IEEE Winter Conference on Applications of Computer Vision, WACV-2015*, pp.580–587, DOI: 10.1109/WACV.2015.83, arXiv: 1510.02078.

Chen, S. et al. (2018) 'MobileFaceNets: efficient CNNs for accurate real-time face verification on mobile devices', *Chinese Conference on Biometric Recognition*, Springer, pp.428–438.

Directorate of Primary Education (2020) *Operational Guideline for School Meal Programme in Bangladesh*, pp. 1–8 [online] http://sfp.dpe.gov.bd (accessed July 2022)

Elsweiler, D. et al. (2015) 'Towards automatic meal plan recommendations for balanced nutrition', *RecSys 2015 – Proceedings of the 9th ACM Conference on Recommender Systems*, pp.313–316, DOI: 10.1145/2792838.2799665.

Garey, M.R. et al. (1979) *Computers and Intractability*, Vol. 174, pp.1–100, W.H. Freeman, San Francisco.

Government of India (2021) *Mid Day Meal Scheme* [online] http://mdm.nic.in/mdm_website/ (accessed 8 July 2021).

Guo, Y. et al. (2016) 'MS-Celeb-1M: a dataset and benchmark for large-scale face recognition', *European Conference on Computer Vision*, Springer, pp.87–102.

Han, J. et al. (2012) 'Getting to know your data', *Data Mining*, pp.39–82, DOI: 10.1016/B978-0-12-381479-1.00002-2.

Haque, F. et al. (2013) 'Outbreak of mass sociogenic illness in a school feeding program in Northwest Bangladesh, 2010', *PLoS ONE*, Vol. 8, No. 11, pp.1–8, ISSN: 1932-6203, DOI: 10.1371/journal.pone.0080420.

Jayakumar, D. et al. (2020) 'Mid day meals scheme monitoring system in school using image processing techniques', *7th International Conference on Smart Structures and Systems, ICSSS 2020*, pp.1–5, DOI: 10.1109/ICSSS49621.2020.9202347.

Jiang, L. et al. (2020) 'DeepFood: food image analysis and dietary assessment via deep model', *IEEE Access*, Vol. 8, pp.47477–47489, DOI: 10.1109/ACCESS.2020.2973625.

Kadari, R. et al. (2016) 'Strengthening the mid-day meal scheme through MIS', *South Asian Journal of Engineering and Technology*, Vol. 2, No. 10, pp.1–9.

Kagaya, H. et al. (2014) 'Food detection and recognition using convolutional neural network', *Proceedings of the 22nd ACM International Conference on Multimedia*, pp.1085–1088.

Kuntagod, N. et al. (2021) 'A framework for enabling safe and resilient food factories for public feeding programs', *2021 IEEE/ACM 3rd International Workshop on Software Engineering for Healthcare (SEH)*, pp.1–4, DOI: 10.1109/SEH52539.2021.00008.

Marrero, A. et al. (2019) 'On the automatic planning of healthy and balanced menus', *Proceedings of the Genetic and Evolutionary Computation Conference Companion (GECCO)*, pp.71–72.

Nukapeyi, S. et al. (2022) 'Implementation of mid-day meal scheme into limelight', *2022 International Conference on Computing, Communication and Power Technology (IC3P)*, pp.279–284, DOI: 10.1109/IC3P52835.2022.00066.

Samal, A. et al. (2019) 'Smart attendance based decision support system for mid-day meal scheme', *Proceedings International Conference on Information Technology*, pp.365–370, DOI: 10.1109/ICIT48102.2019.00071.

Schroff, F. et al. (2015) 'FaceNet: a unified embedding for face recognition and clustering', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.815–823.

Segredo, E. et al. (2020) 'SCHOOLTHY: automatic menu planner for healthy and balanced school meals', *IEEE Access*, Vol. 8, pp.113200–113218, DOI: 10.1109/ACCESS. 2020.3003067.

Simonyan, K. and Zisserman, A. (2015) 'Very deep convolutional networks for large-scale image recognition', *3rd International Conference on Learning Representations, ICLR-2015*, pp.1–14.

Tan, M. et al. (2019) 'EfficientNet: rethinking model scaling for convolutional neural networks', *International Conference on Machine Learning, ICML 2019*, Long Beach, pp.6105–6114.

Tan, M. et al. (2020) 'EfficientDet: scalable and efficient object detection', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.10781–10790.

Viola, P. et al. (2004) 'Robust real-time face detection', *International Journal of Computer Vision*, Vol. 57, No. 2, pp.137–154.

World Health Organization (2021) *Malnutrition*, June, pp.1–2, https://www.who.int/news-room/fact-sheets/detail/malnutrition (accessed 20 February 2022).