Tetris Team
Smart Driver Assistant

- MEHMET KURHAN
- NAIL IBRAHIMLI
- SEYMUR MAMMADLI
- SHKELQIM MEMOLLA

Outline

- Introduction

- Application architecture

- UI Design

- Features & Methods

- References

Introduction

▶ *Problem ?*

The purpose of this project is to reduce the number of car accidents caused by drivers only because of their phone usage and the cases when they are feeling sleepy.

Smart Driver Assistant takes the phone and uses it in a way that serves as a guard of the driver and gives the driver the opportunity to be fully focused and have a high probability of avoiding accidents.

Application architecture
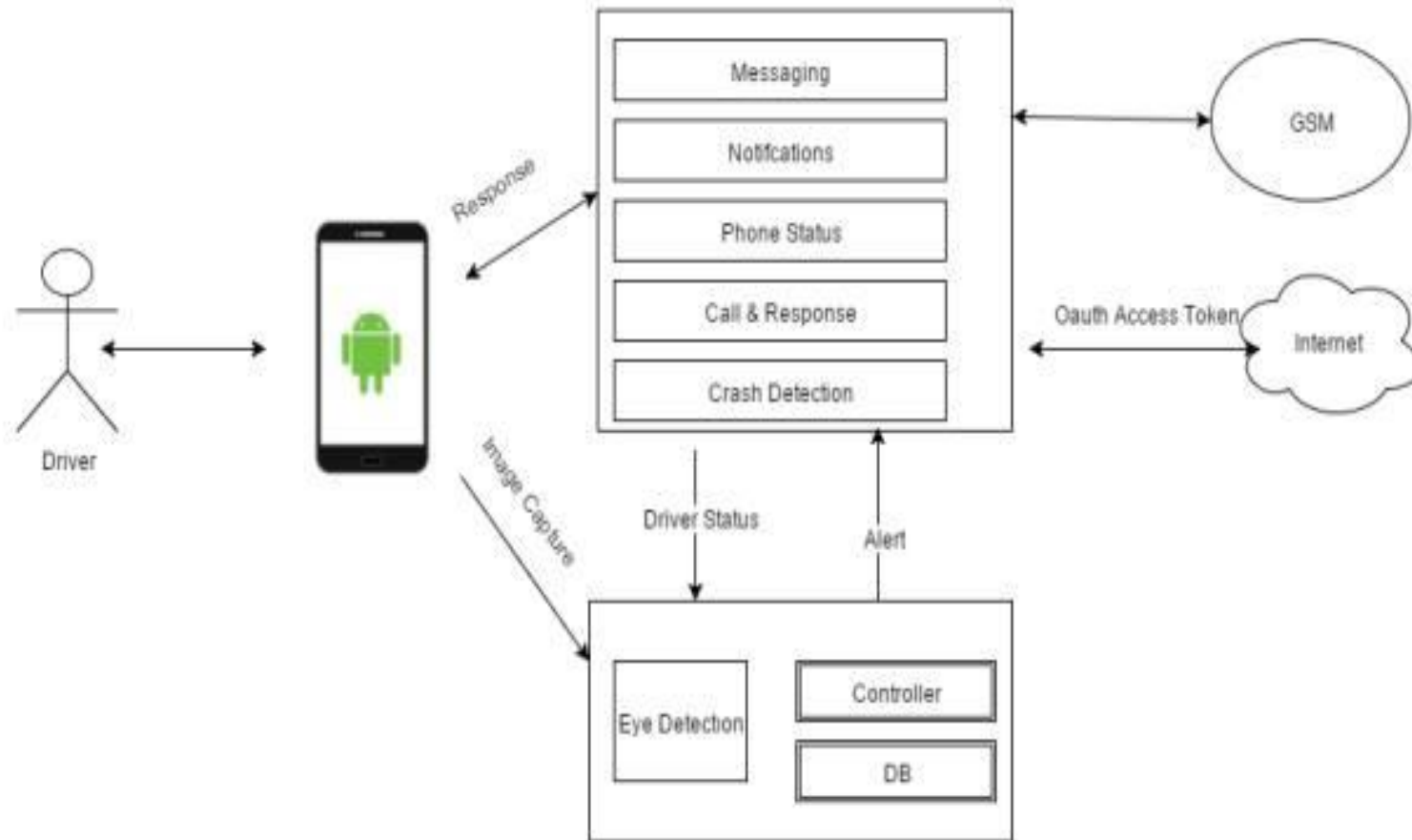
UI Design

Main activity contains four buttons

- **VOICE COMMANDS**

This button starts voice recognition service as a background task.
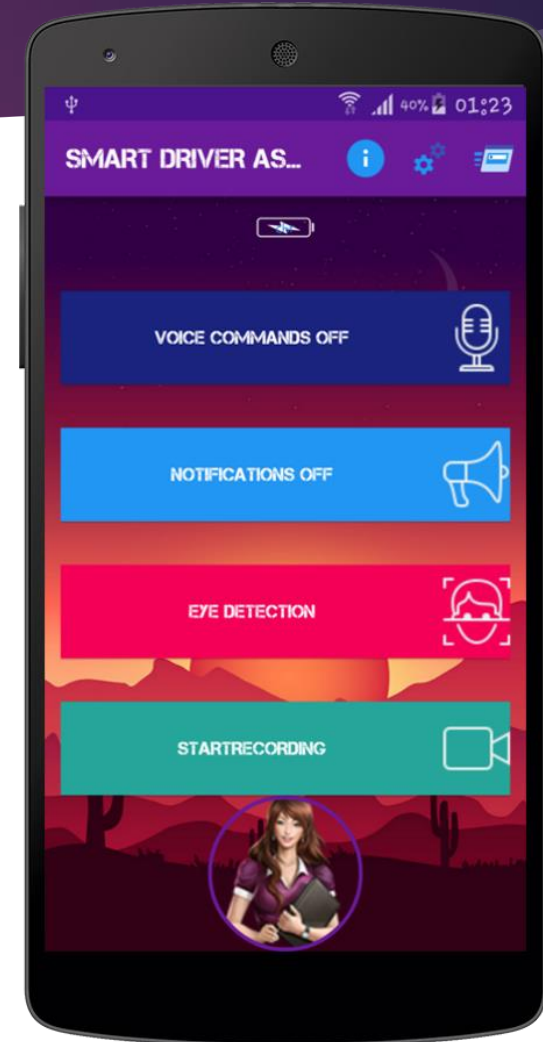
- **NOTIFICATIONS**

Notification service is activated. API 17 or higher is required.
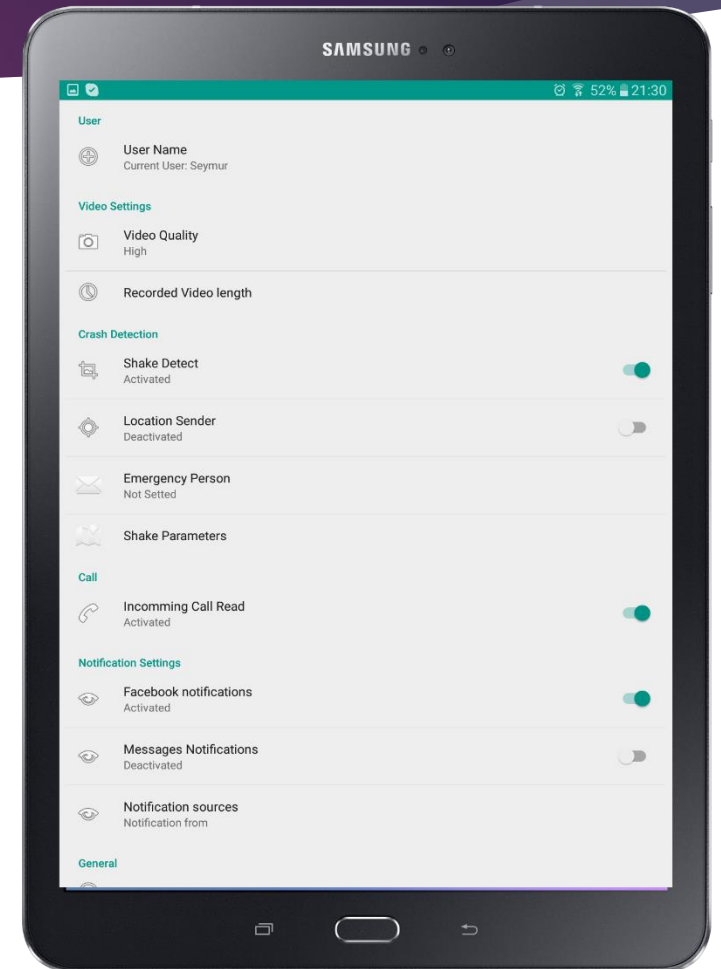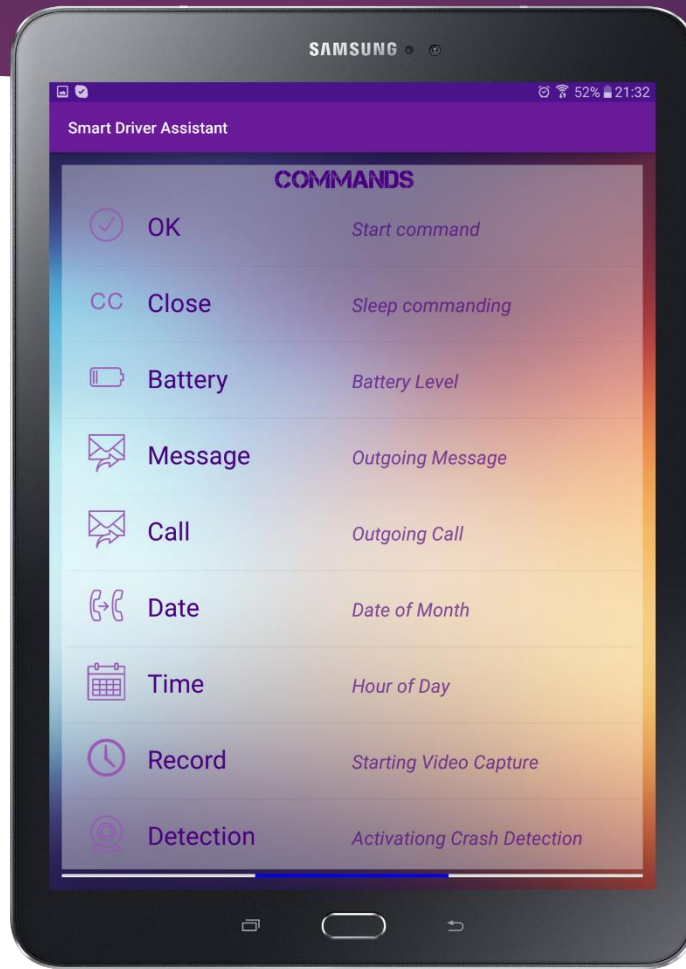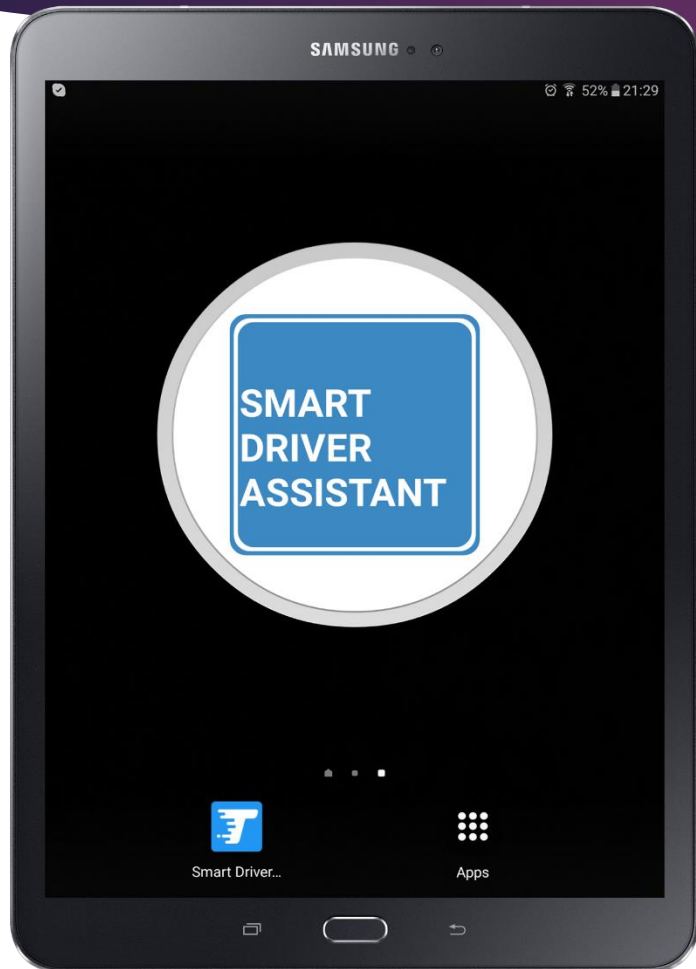
- **EYE DETECTION**

Drowsy driver monitoring will start as a new activity.

- **STARTRECORDING**

Recording with a infinite time interval is initiated. This task also runs as a background service.

UI Design

Features & Methods

▶ Voice Command

This feature uses Android Speech To Text and Text To Speech libraries. Speech recognizer runs as a background service and no physical interaction is required. Multiple languages are supported.

▶ Call

❑ The application interact with the user, by asking the contact name, wait for confirmation and successfully performs the call process

▶ Message

❑ User first creates a short message, application asks the user for message confirmation and then for contact name. Under user's approval message is sent

Features & Methods

▶ Voice Command

  ▶ Record

   ❑ The user can activate video recording feature with voice. It supports adjustments for choosing time interval and quality.

  ▶ Detect

   ❑ The user can activate drowsiness detection feature.

  ▶ Time
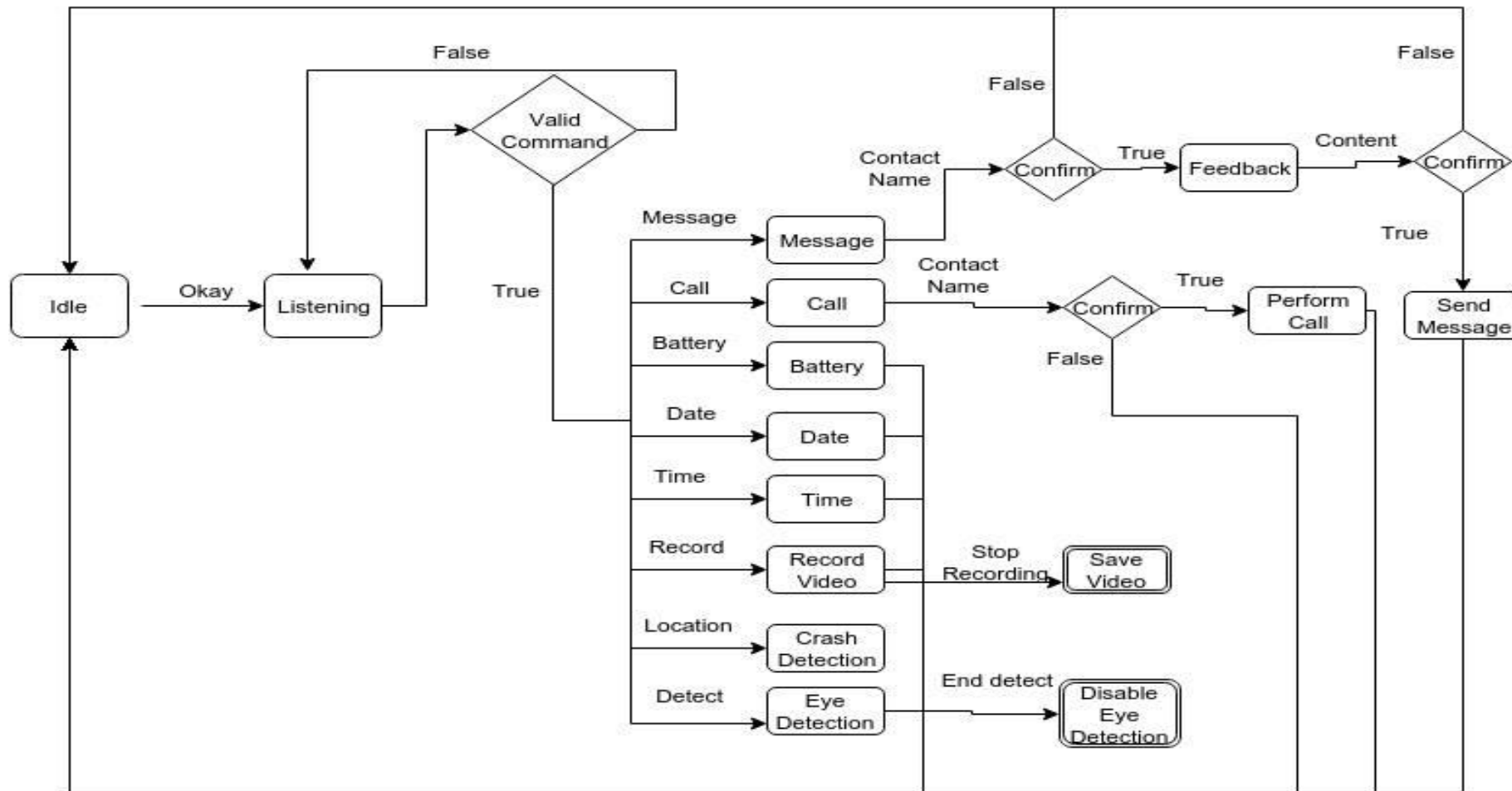
   ❑ Current date and time are read to the user.

  ▶ Phone Status

   ❑ Information about current status of the device like battery level is read.

Voice Command's flow diagram

Features & Methods

▶ Notification Handler

Incoming notification is read loudly to the user. For performing this feature Android Notification Listener service is used.

- ❑ Notification coming from Facebook is directly read to the user
- ❑ Notification coming from GSM phone is read to the user and also displayed on the screen
- ❑ User is also informed about arrival of notification from other sources such as WhatsApp, Gmail, …

Features & Methods

▶ Car Crash Detection

This feature is implemented using device's built-in accelerometer sensor. It detects extra forces and duration of that force on the device.

If the detected force is greater than an adjusted threshold, application asks driver's feedback and if driver answer is negative or there is no answer, driver's current location information such as city and street name are sent to Emergency contact. Force parameters such as minimum amount of direction change, duration, minimum value are adjustable.

Features & Methods

▶ Driver Drowsiness Detection

This feature is implemented using both Google Android Vision library and OpenCV library. From OpenCV, HaarCascade method is used.

Both methods runs on separate threads, with the same frame. And if one of the methods detect eyes being closed, the application rings the alarm.

References

- [1] http://opencv.org/platforms/android.html

- [2] https://developers.google.com/vision/