EYES - UP

APPLICATION

PRESENTED BY:

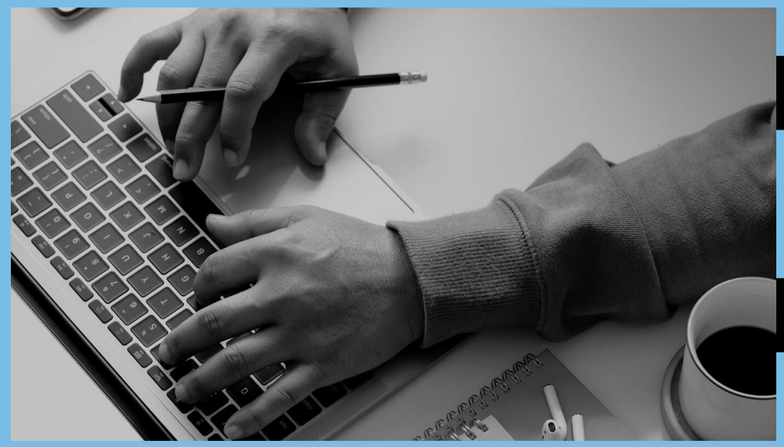
AHLAM ABUATLLAH 1706758
MAY ALREFAE 1705094
MSHAER ALZUBAIDI 1708032

Ahlam Abuatallah

1706758

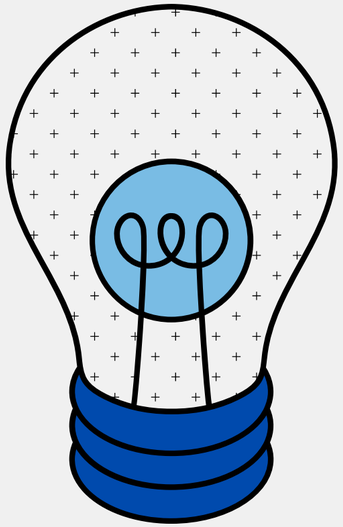
TOPICS

- PROBLEM
- SOLUTION
- FEATURES
- DESIGN OF SYSTEM
- TESTING & REULTS
- PROJECT LIMITATION
- FUTURE WORK
- CONCLUSION



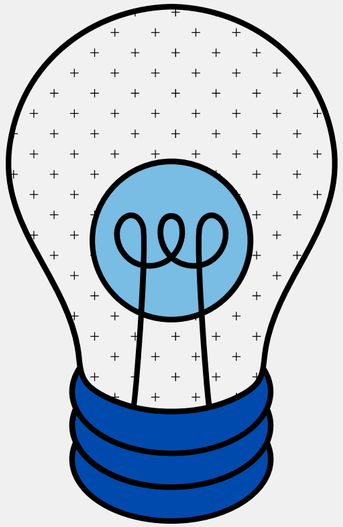
PROBLEM

Every driver fear is getting into an accident and putting their loved ones in a bad situation, missing signs or get distracted from the road which often result in accidents and serious injuries, and the worst fear of all is getting sleepy while driving and start drifting from the lane or being unable to keep your eyes open.

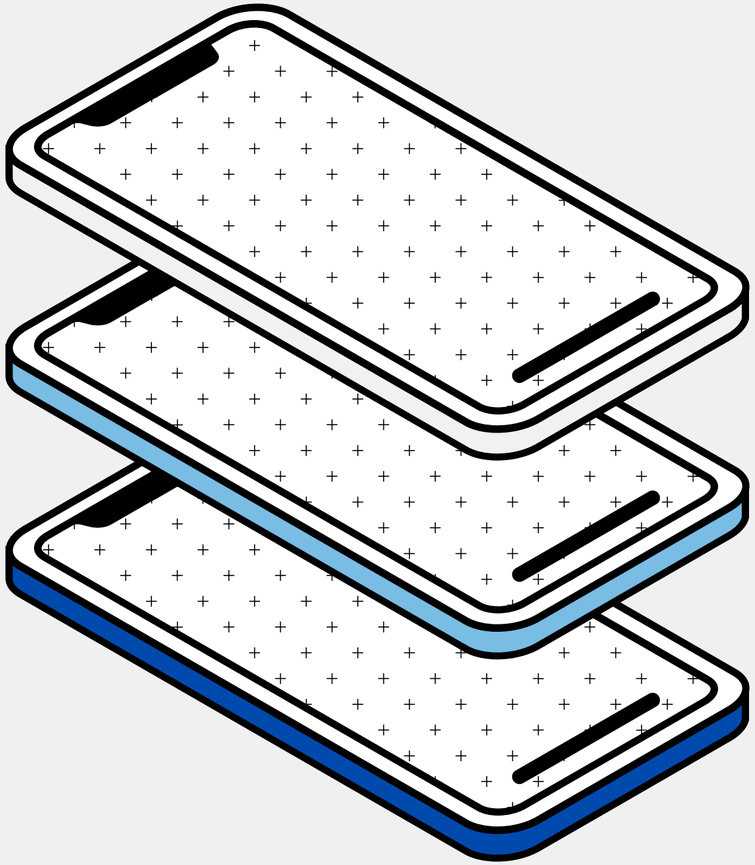


SOLUTION

Our application will provide a safer driving experience for all drivers. Eyes-Up will use the facial recognition technology to detect eyes, if users closed their eyes alerts will go off to notify the driver.



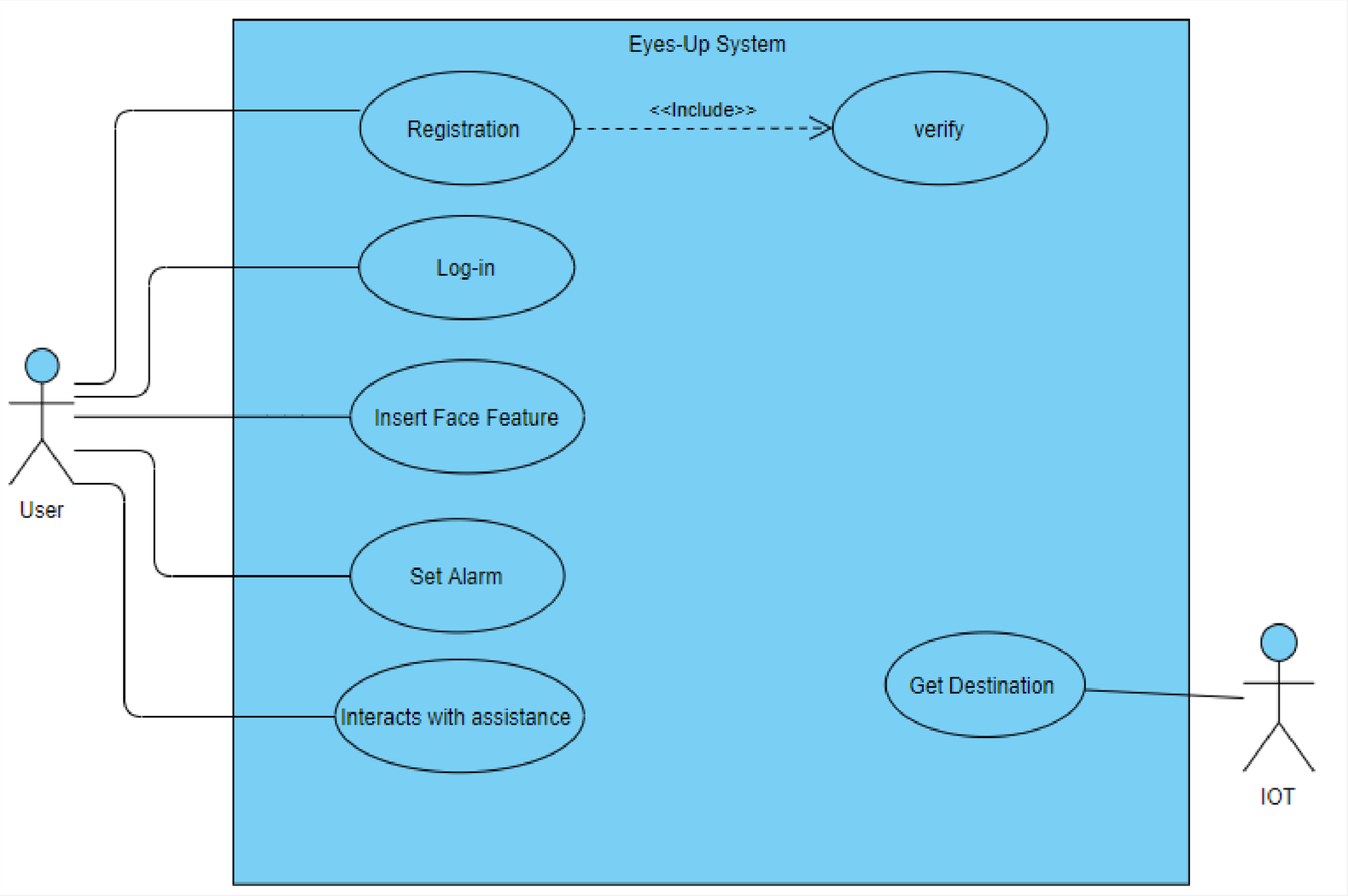
FEATURES



- Face recognition in our application will be accurate and will detect closed eyes for 9 seconds before the first alarm is activated, second alarm will start after 21 seconds asking the user to answer by voice that they are awake last alarm will start after 30 seconds calling the emergency contact
- You can customize your own alarm whether it's text message notification or sound alarm and set it when the application detects your drowsiness.
- Allow the application to call your emergency contact in case of emergency.
- It will also show the driver to the nearest café when drowsiness is detected.
- User can enter any notes they need to remember to do along the way

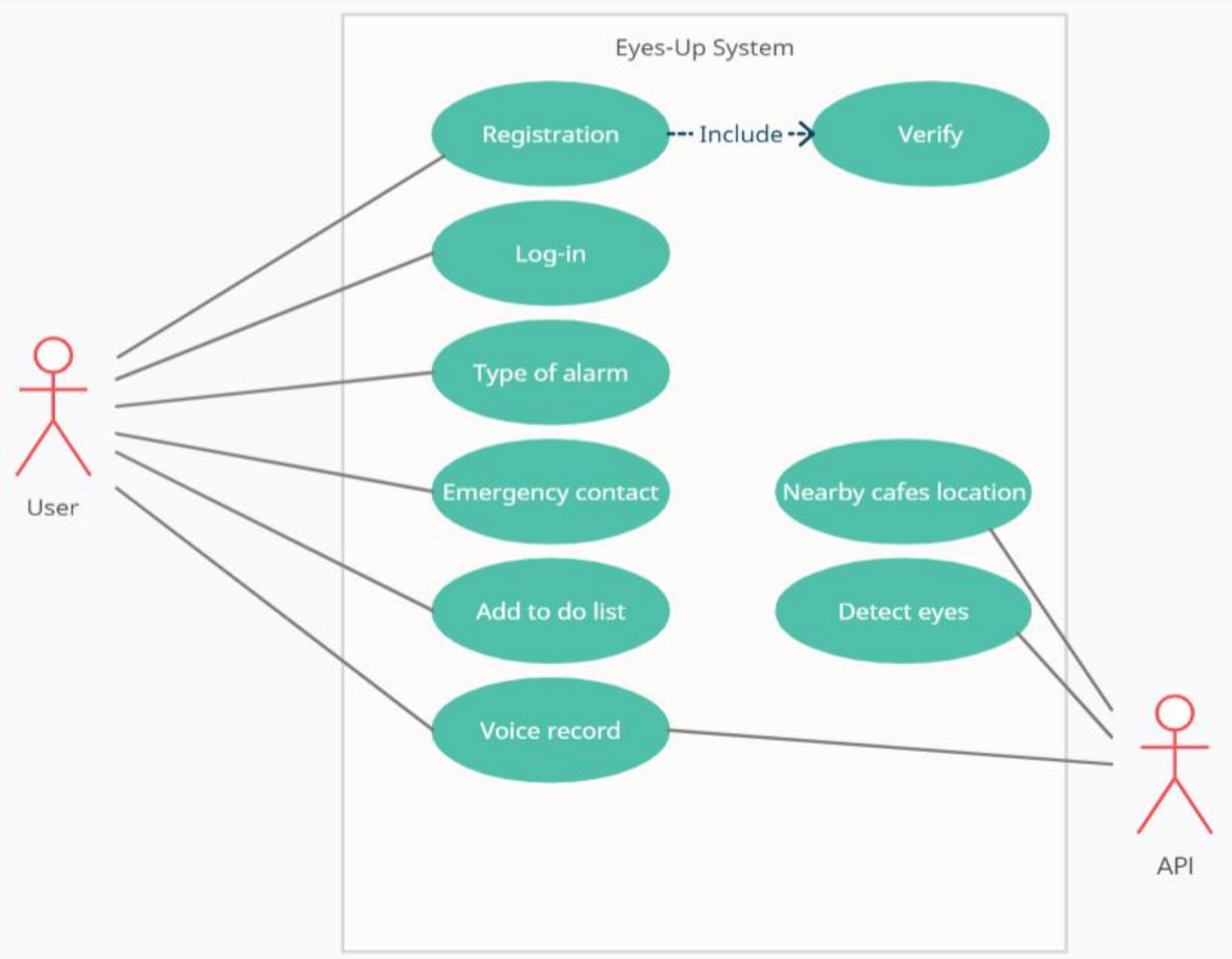
Design of System

Old Use Case



Design of System

New Use Case

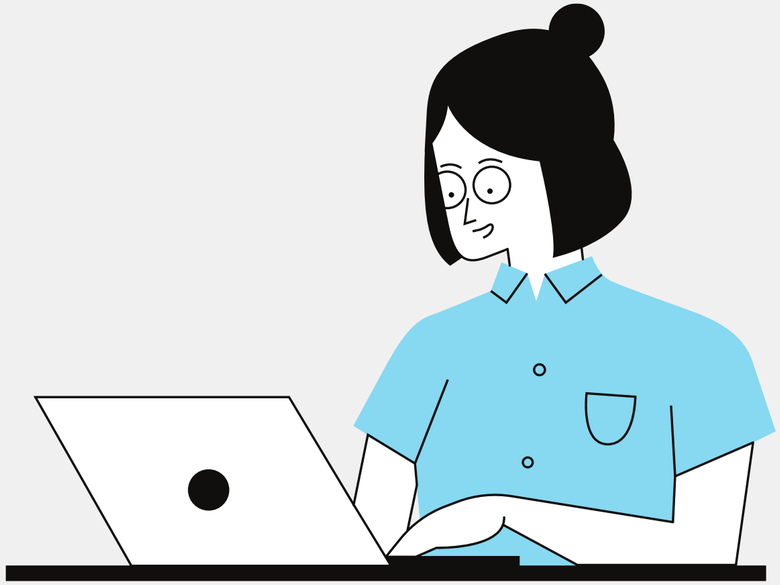


Mshaer Alzubaidi

1708032

TESTING

Three types of system testing is performed on the Eyes-Up application.



1. System testing.

System testing tests the application, and it is done manually by trying different test scenarios and observing the application results

2. Unit testing

Unit testing is a type of testing in which Its primary purpose is to test different unit or function.

3. Useability testing

Useability testing Refers to how easy users find it to accomplish specific function

Testing & Result

System testing

Test scenario	Result
login->home -> menu -> note	pass
login-> faces detect -> sound alarm -> nearby cafe	pass
login-> account page->update information -> sign out	pass
login-> face detect -> pop up message alarm -> press the message button	pass
login-> menu -> change alarm type -> pop up message	pass
login-> menu -> change alarm type - > sound alarm	pass

Testing & Result

Unit testing

Functionality	Input	Expected result	Result
Check Password	More than 8 - At least one small letter At least one capital letter - No special character	True	Pass
	less than 8 - At least one small letter At least one capital letter - No special character	False	Pass
	More then 8 - No small letter At least one capital letter - No special character	False	Pass
	More then 8 - At least one small letter No capital letter - No special character	False	Pass
	More then 8 - At least one small letter At least one capital letter - with special character	False	Pass

Testing & Result

Unit testing

```
@Test
public void testcheckPasswod(){
    System.out.println("checkPasswod");
    String password = "1234567Cd";
    User instance = new User();
    boolean expResult = true;
    boolean result = instance.checkPasswod(password);
    assertEquals(expResult, result);
}
```

Testing & Result

Unit testing

Functionality	Input	Expected result	Result
Check phone number	More than 10 No letters	True	Pass
	less than 10 No letters	False	Pass
	More than 10 With letters	False	Pass

Testing & Result

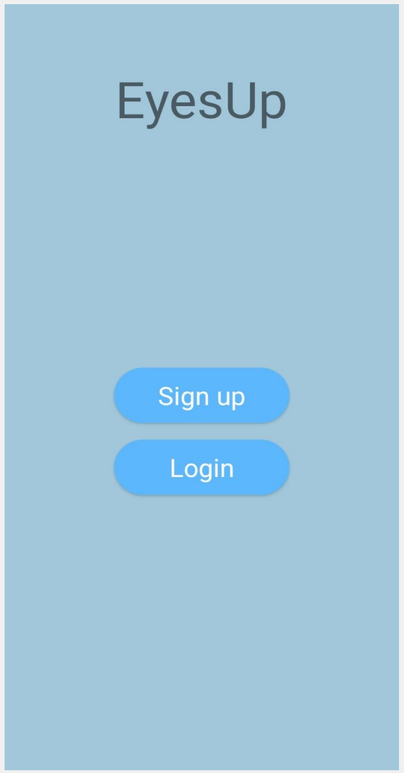
Unit testing

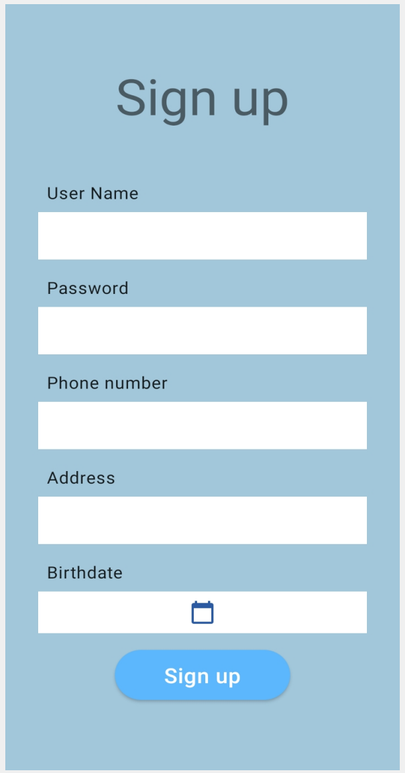
```java
@Test
public void testcheckPhone(){
    System.out.println("checkPhone");
    String phone = "1234567890";
    User instance = new User();
    boolean expResult = true;
    boolean result = instance.checkPhone(phone);
    assertEquals(expResult, result);
}
```

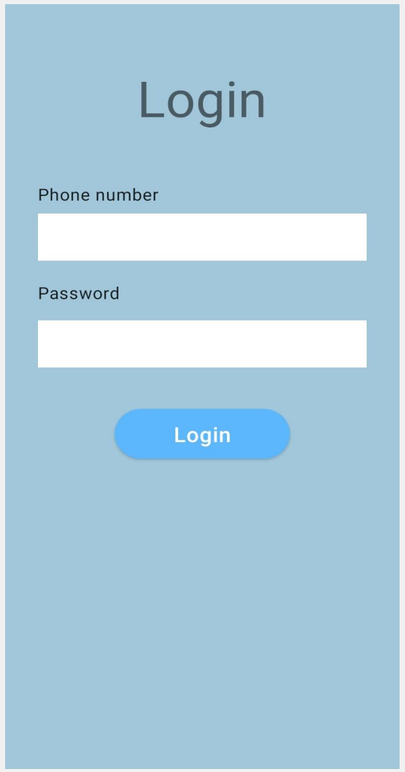
May Alrefaee

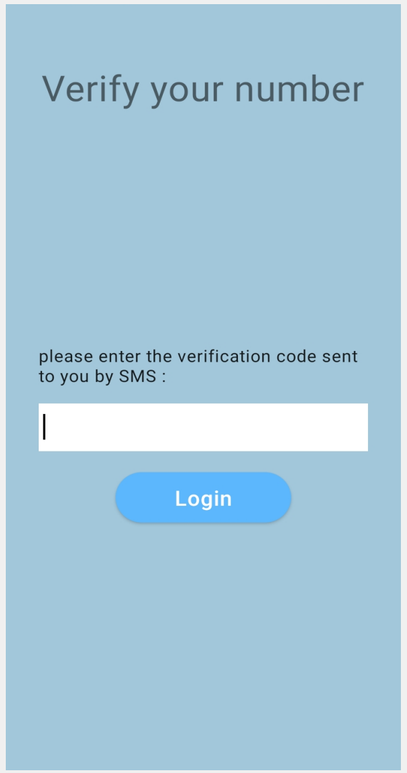
1705094

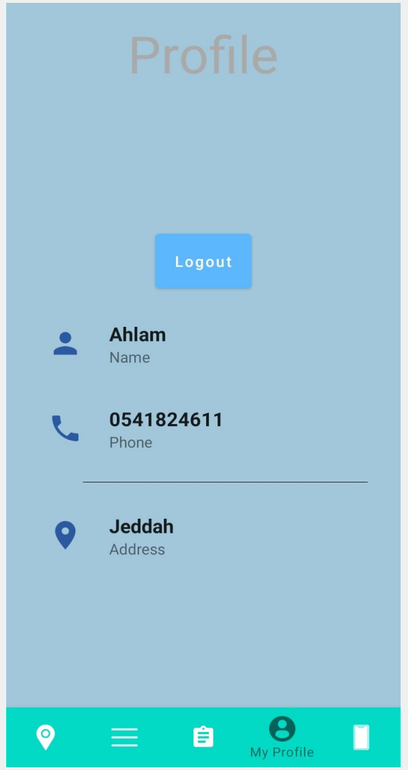
PROJECT RESULT





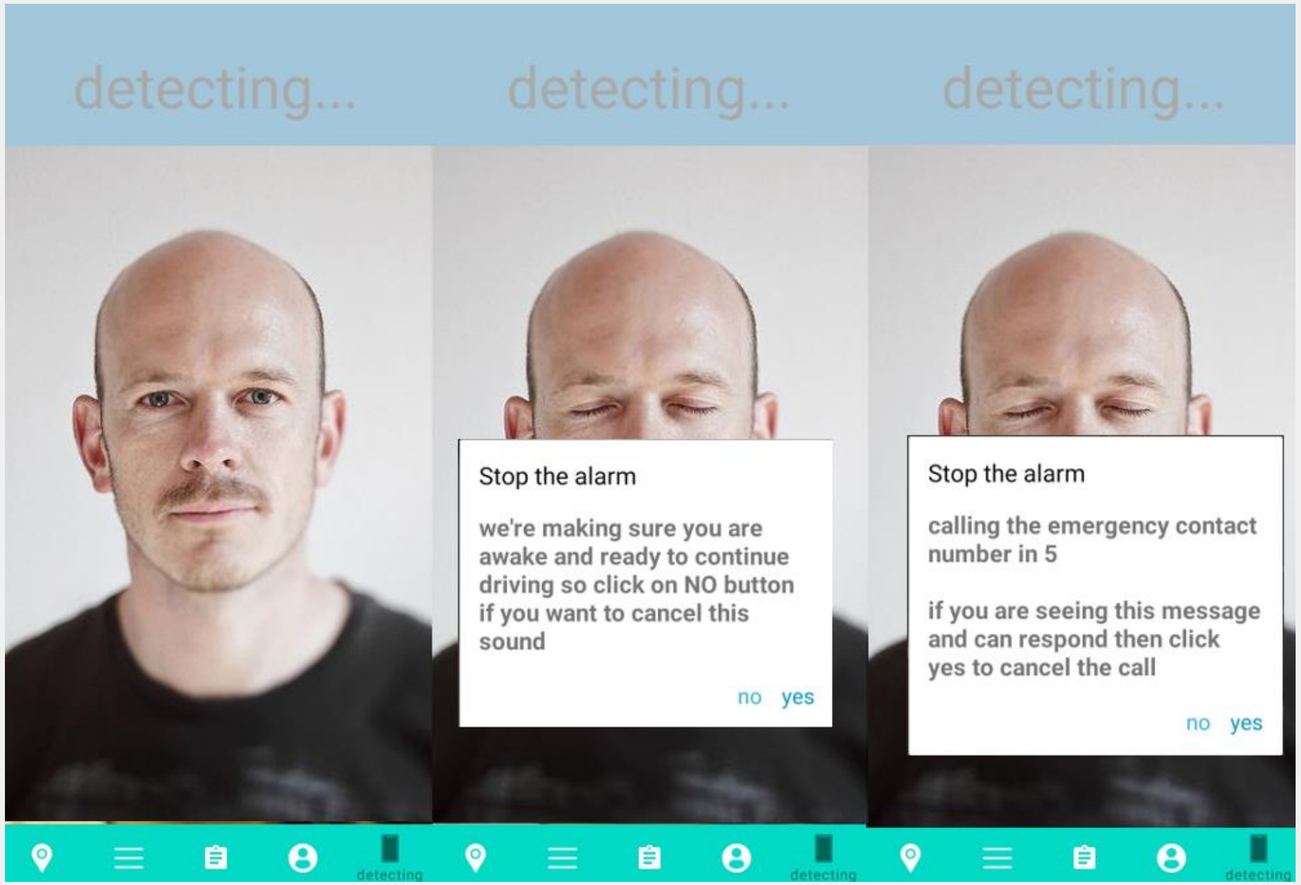


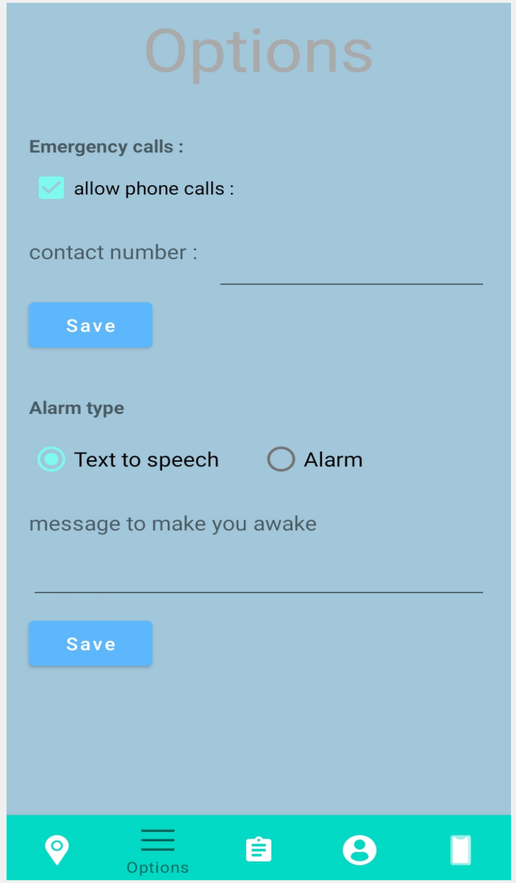




Here are the application results starting with sign up, login page, verifcation and profile page

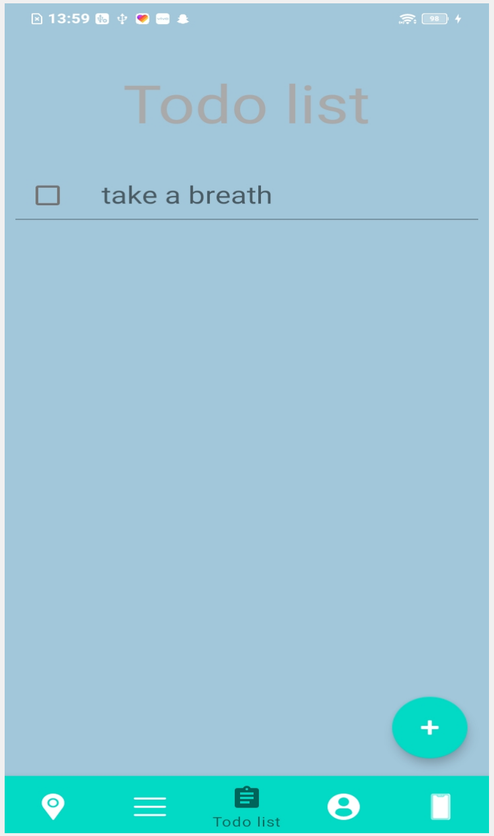
PROJECT RESULT

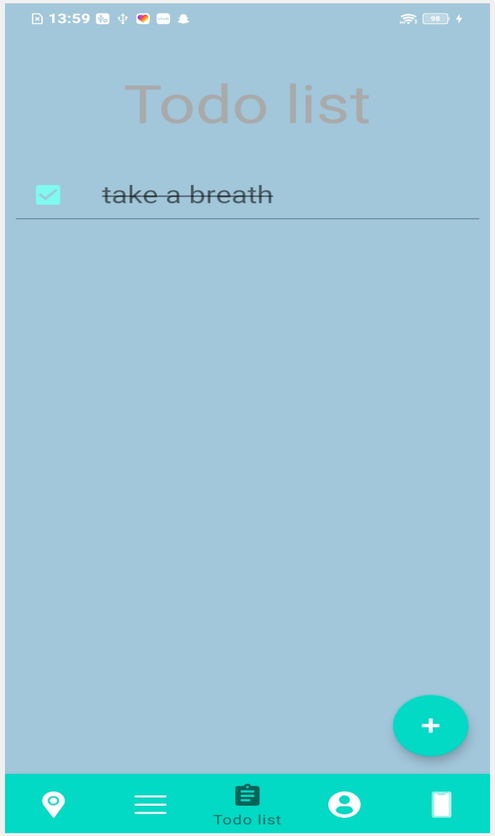




Here is how the application detect the drowsiness and the option page that can customize the alarm. furthermore, the map showing the nearest cafes

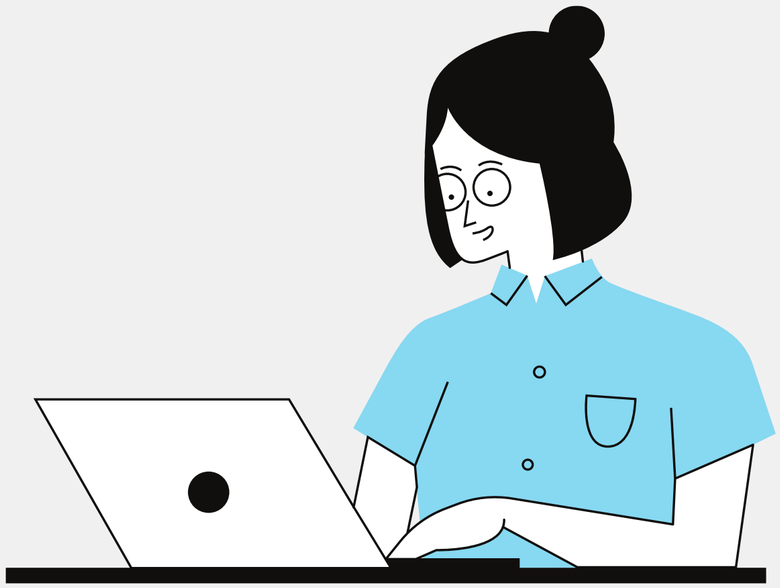
PROJECT RESULT





this page the user can enter any notes they need to remember to do along the way, and when they are done with, they can just check the box.

PROJECT LIMITAION



1. THE LOGIN VERIFICATION

we could not do it because it needs another system included and it was expensive to purchase.

2. THE SMART WATCH

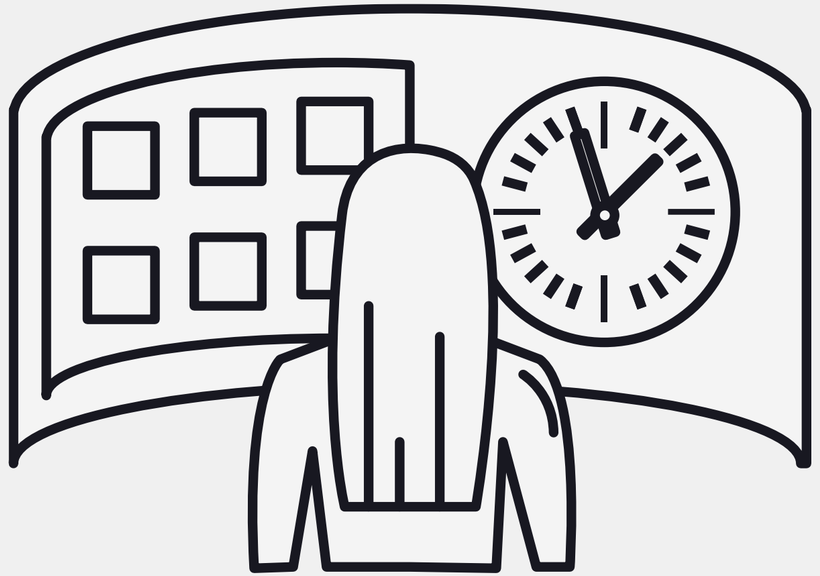
connecting with the application to calculate the driver's heartbeat to check if they are sleepy but we could not do it because of the limitation of time.

3. CALL 911

we did not force the application to call 911 incase of emergency but will let the driver choose who they want the application to call for them.

FUTURE WORK

We will work on developing the overall performance of the application such as the interfaces, the timer of capturing the face also capturing the yawing of the user for detecting the drowsiness, we will also work on connecting our application to smart watch to increase the features.



CONCLUSION

Eyes-Up application is an efficient solution to safer driving experience for all drivers. we made sure it works properly as described, with face detection and speech recognition.

THANK YOU!

Any Questions?

Ahlam Abuatllah
May Alrefae
Mshaer Alzubaidi