

Computer Games Development CW208

SRS and Project Report

Year IV

|  |
| --- |
| Sean Abner Nash de Andrade |
| C00217019 |
|  |
|  |
|  |
|  |

# 



**Faculty of Science**

**Open-Book and Remote Assessment Cover Page**

**Student Name: Seán A’bner Nash De Andrade**

**Student Number: C00217019**

**Lecturer Name: Martin Harrigan**

**Module: Project II (Fyp)**

**Stage/Year: Fourth Year**

**Date: 03/05/2020**

**Declaration**

This examination/assessment will be submitted using Turnitin as the online submission tool. By submitting my examination/assessment to Turnitin, I am declaring that this examination/assessment is my own work. I understand that I may be required to orally defend any of my answers, to the lecturer, at a given time after the examination/assessment has been completed, as outlined in the student regulations.

# Table of Contents

[**Table of Contents**](#_heading=h.7gpfftrjy67e) **1**

[**Project Abstract**](#_heading=h.9n6mtmmoni9r) **2**

[**Project Introduction**](#_heading=h.c4fjzyomdsxs) **2**

[**Background**](#_heading=h.sh278dl465t9) **3**

[**Project Description**](#_heading=h.sfxtgmmfrwyn) **3**

[**Technical Achievements**](#_heading=h.6a58kv64z0ty) **3**

[**Personal Achievements**](#_heading=h.1cilddbu13oo) **4**

[**Overview**](#_heading=h.tyjcwt) **4**

[**Philosophy**](#_heading=h.9hpk6oqan2p9) **4**

[**Define the Application**](#_heading=h.3dy6vkm) **5**

[**What is the application supposed to do**](#_heading=h.k2wo4x149ziq) **8**

[**Who is going to be using this application**](#_heading=h.kjy1qsd5n75j) **8**

[**Context Diagram and Use Cases**](#_heading=h.eun6rdezqlwr) **9**

[Capture Test Result](#_heading=h.t9lacnw2mud7) 9

[**Metrics**](#_heading=h.y9day14iig7e) **9**

[**Is there a precedent for this application?**](#_heading=h.xmn9fgrqfuw7) **9**

[**Feature List**](#_heading=h.x6txmzsvcz2i) **9**

[Feature 1: Result Block](#_heading=h.efglq9pm12f) 9

[Feature 2: React Frontend](#_heading=h.hx1uwoknt0fs) 10

[Feature 3: Home Screen](#_heading=h.3jdca4nkzic6) 10

[Feature 4: Camera Screen](#_heading=h.fs7tdvkdzlgd) 10

[Feature 6: Result Screen](#_heading=h.4statcyqaz1n) 10

# 

# Project Abstract

This project is part of the Erasmus+ Programme. It uses the following Technology and code: Pytorch, Python and React. I will be creating a React Native front end application to allow a user to capture the results on a GIP Test Strick (Gluten Immunogenic Peptides). This stick indicates if a person has consumed gluten and if it is present in their system. This image will then be sent to a server backend where a photo recognition model will return a result based on what result it believes the picture to show. This result will then be displayed to the user in an clear and concise fashion in the UI.The date and time are displayed as is the result and accuracy of said result. The user is able to go through their entire history of tests with minimal effort.

# Project Introduction

This project is part of the Erasmus+ programme and its title is DESQOL which is a pseudo-acronym for “Teaching Interdisciplinary Human Centred Design to Improve Patient Quality of Life”. This programme has participants from the following:

1. IT Carlow

2. FHV, Vorarlberg University of Applied Sciences

3. HvA, Amsterdam

4. St. Luke's General Hospital Carlow-Kilkenny, HSE

5. Grupo IHP, Andalucia, Spain

Research has shown that teenagers with Coeliac’s Disease tend to struggle to comply with their dietary requirements. To aid them in meeting their gluten free diet requirement the above partners are working to create an application that will help comply with their diet and make their lives easier. The target age group is the 13 - 18 range. Compliant people tend to maintain their gluten free diets but Non-compliant people tend to fail to avoid Gluten constantly, as this behaviour can lead to type-1 Diabetes in the long run methods to help avoid this are needed.  
  
A meeting in Carlow had a few ideas being discussed, ranging from a VR game to teach Celiac patients how to safely prepare food without contamination or which foods are safe to eat to GIP Sticks. A GIP stick kit costs about 12 Euro, Urine can be tested half an hour after ingestion to check for the presence of Gluten. However, results disappear after 20 hours. Faeces tests are done one or two days after ingestion and its results will last for four days before disappearing.

These sticks give either positive or negative results. Lab equipment is more accurate but the purpose of these tests is not to check for Gluten but to monitor intake and compliance. To give certainty to the patients that they are dealing with things properly. The test takes between ten to fifteen minutes depending on what matter is being tested.

# Background

I’m taking inspiration from previous Celiapp projects and from various photo adjustment apps that are commonly around. They focus on quickly allowing the user to reach their core functionality and simple UI to minimize initial learning time.

I have no real experience with machine learning other than a few dabbling attempts to create a neural network AI. I have been doing some research into google’s TensorFlow and Facebook’s PyTorch. There were various discussions about the better option for which situation.   
  
I have also been looking into the details of UI design and the minor effects that make UI more effective for the user’s end goal.

# Project Description

My app is a part of the Erasmus+ App. It’s goal is to allow Celiac patients to take a picture of their GIP test strips to get the results read. This is achieved through the use of a deep learning model trained on the sticks which will read the received photos and output a result along with a percentage. My contribution to this project will be constructing the front end of the application that users will use. This will allow them to easily capture images of their tests and read through their result history easily. The application will user react and will be intergrated into the server backend of my predecessors photo recognition model.

## Technical Achievements

I have never worked with or created a Machine Learning system before or made a React front end. Using these involves quite a bit of research, relearning habits and adapting to new logic paths. I feel that improving the accuracy and speed of the results from the model and allowing a smooth transition from taking a picture to receiving the results is the ideal goal.

## Personal Achievements

I have learned a lot about Machine Learning. Pytorch and Tensorflow in particular but the field in general as well during my research into the topic. I’ve always been interested in the field but I do feel I should be able to make progress in this area in the future should I need to specialize. React while providing several design challenges and new ways of building front end pages does offer valuable experience. Many one page apps and sites will likely make use of this system in the future and getting experience ahead of time is quite useful. I’ve also never designed and made an app from the ground up before. Doing this has been quite enlightening about the necessary steps and planning needed along the way.

# Overview

## Philosophy

Providing Certainty

Fundamentally what I feel Celiac patients might deal with, especially newly diagnosed ones is a sense of unease based on lack of knowledge. They don’t know exactly what parts of their life are fine and what needs to be redone from the ground up. What I aim to achieve with my app is provide some confidence to them. The GIP sticks are useful but due to chemistry being a bit unreliable it can be hard to come up with a proper result. Especially with untrained eyes. I feel that the app and the tests in conjunction will be a good sign to the patients when they are doing well, and an indicator for when something has gone slightly wrong and they should keep an eye on their recent intake.

Encouragement

The second aim for the app is to provide an incentive, the more reason for a compliant patient to keep doing so and reduce the number of non compliant patients. This is done by encouraging them to take up a habit which encourages reflection on recent actions. Thinking back on food that might contain gluten and possible avenues of contamination. I feel that continual signs are important to keep users informed, doing so helps keep them involved.

Phones and The Target Audience

The App will be based in React and be available on Phones and IOS devices. The reason for this being that the main target audience is young people between 13 and 21. The ages when they start to get some independence from their caretakers and can choose their own food on time. When they’re learning what they need to do to take care of themselves in the future and when they’re most vulnerable to peer pressure. It’s easy to imagine a patient just deciding to take the discomfort to spend some time with their friends at a convenient fast food place. But repetition is how a habit is made and this one will harm their health in the long run. As near all of them have phones, are capable of using them and tend to have them within arms reach at all times this device is the ideal platform for when they’re looking for answers and information.

# Define the Application

In short the application will allow the user to take pictures of their GIP stick test and it will tell them the results. In length it should function like this.

Main Page.

The user should select the app from their phone as normal. Once the app opens it displays the main screen. The user has few options but there is room for more options to be added should the Application be combined with other components from other projects. For now there will be two buttons, labelled “Camera” and “Results”. The user can also exit out from the app like other apps through the use of the home button.

The User can from here select either of the available options. There should be a sound effect feedback on clicking a button. Each button will bring the user to the indicated page.  
  
 Mock Up Home Screen

# 

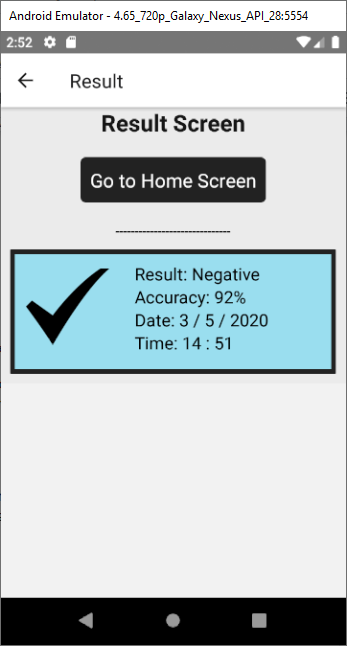
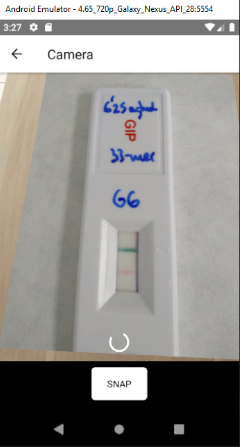
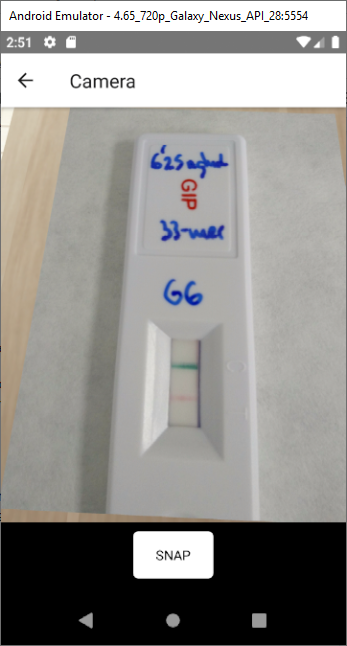
Camera Screen.

Upon pressing the Camera button on the main menu the user is brought to the camera screen. This screen should access and show the user what the camera on the back of the phone can view. From here the user can read on screen instructions to point the camera at the GIP stick and press the screen to take a picture. This picture is immediately passed into the Photo Recognition Model and a result is calculated. During this period a loading screen is shown if needed. Once a result is calculated, The user is brought to the result screen. From here the user can go back to the Camera screen or back to the homescreen.

From the Camera Screen the User can go back to the Home Screen.

Mock Up

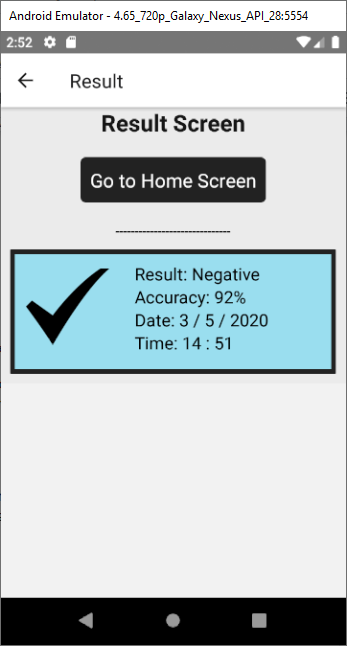
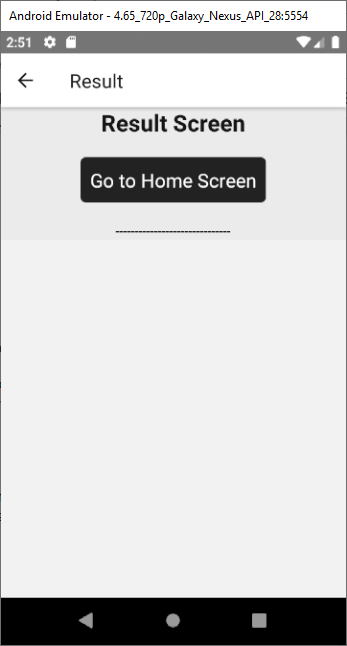
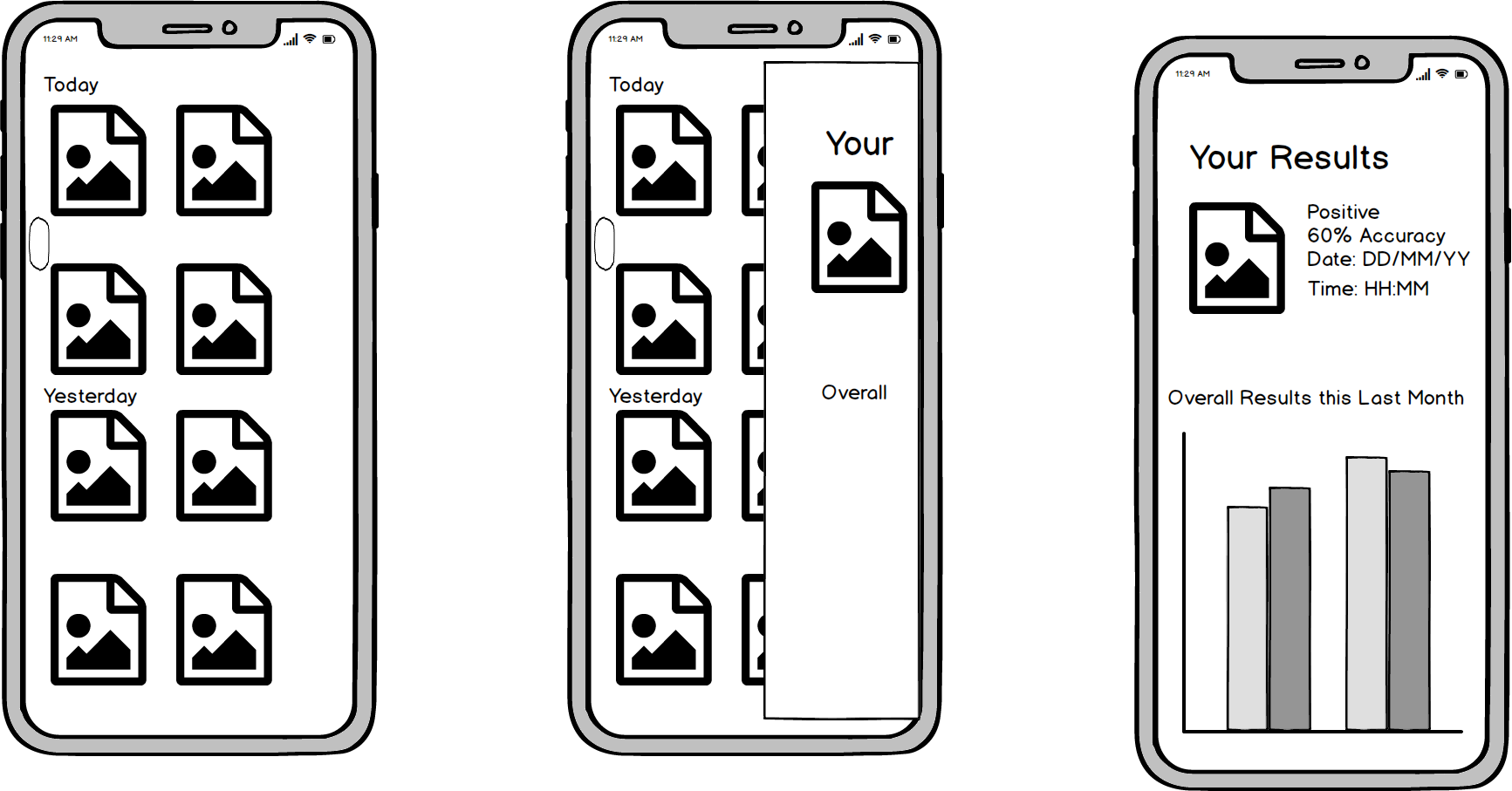
# 



Result Screen.

When the User selects the Result button from the Home screen they are brought to the Result screen. They are shown widgets of previous tests they have taken along with dates. The tests are ordered by time taken.The phone’s Date and Time settings are used to set these.

The Thumbnails are arranged in rows, with a scroll bar indicating to the user that they can swipe up and down to scroll through the history. They will show what result the Photo recognition model thought the tests pictures were showing along with the accuracy of that result along with the date and time the test was taken.

From here the User can go back to the Home Screen.  
  


# What is the application supposed to do

The application is a front end to allow the user to easily capture their GIP tests and send it to a server to get confirmation on the results along with allowing them to easily look over their history of tests to see if there is a trend occurring with their test results.

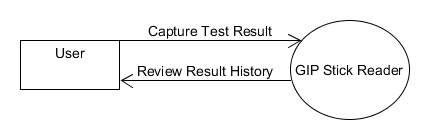
The application aims to be a user-friendly experience and understandable by everyone even those who are unfamiliar with phone apps.

It should also offer the ability to review previous tests along with a simple graph system to make the data understandable in an instant.

# Who is going to be using this application

Young people between the ages 13 -21 and older.

# Context Diagram and Use Cases



## Capture Test Result

A User wants to confirm the results of their GIP stick. They open up the GIP Stick Reader app on their phone. They are on the home screen and see two buttons. They press the one that says camera. This brings them to the Camera screen where they can see that what is in the back camera’s view is being shown to them along with a button to take a picture.

They point the camera at their GIP stick and press the button to take a picture. A spinning loading widget appears and after a few seconds they are brought to the Result screen. They can see the result of their test on the screen.

# Metrics

The success of the application will be based on the speed at which the model returns results. It’s accuracy in doing so and the number of patients who continue using the app after the 21 day challenge.

# Is there a precedent for this application?

There was a previous app that did something similar. But it had a simple front end. I will be making use of this developed back end server and working on creating a more user friendly front end application using React Native for this project.

# Feature List

## Feature 1: Result Block

Description:

A react native component used to display the results received from the Photo Recognition model on the Server.

Conditions Of Satisfaction:

* Icons should match the result received from the server.
* A too low accuracy will cause the result to be inconclusive.
* The Correct date and time will be shown.

## Feature 2: React Frontend

Description:

The App’s frontend should use React Native.

Conditions Of Satisfaction:

* There should be quick transitions and data feedback.

## Feature 3: Home Screen

Description:

The Screen the User is shown after opening the App. From here they may navigate to the Camera Screen or the Result Screen or quit out of the App.

Conditions Of Satisfaction:

* Both buttons transition to the appropriate Screens.
* Appropriate fonts, Icons and Colours are used.
* It is clear what button brings the user to which screen.

## Feature 4: Camera Screen

Description:

The screen from which the user can take pictures of their GIP tests.

Conditions Of Satisfaction:

* The correct camera should be accessed.
* Pictures can be taken with a finger tap.
* The user can press the back button to go to the Home Screen.
* Taking a picture brings the user directly to the Result Screen.

## Feature 6: Result Screen

Description:

The screen that displays past tests, allows the user to see them all easily and with little effort.

Conditions Of Satisfaction:

* The Tests should be grouped into order.
* The Result screen should be operable only by touch and swipes.
* The User can Return back to the Home Screen.