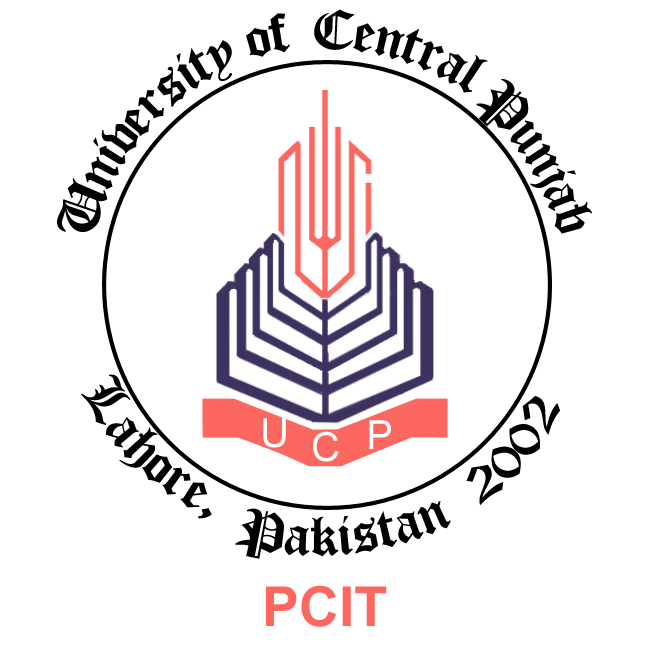
**BSCS FINAL PROJECT PROPOSAL**

Virtual Gym Trainer

*Term of Registration: Fall 2019*



Presented by:

|  |  |
| --- | --- |
| **Registration No:** | **Name:** |
| L1F01BSCS0294 | KHIZER HAMEED |
| L1F01BSCS0241 | SHAHZAD MIRAJ |
| L1F01BSCS0502 | ZAID BIN MOAZ |

|  |
| --- |
| Faculty of Information Technology |

University of Central Punjab

**Project Title**

A workout application with user supervision

**Project Advisor**

Sir Imran Arshad

**Particulars of the students:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Registration#**  eg.**L1F00BSCS0101** | **Name in Full**  Use Block Letters | **CGPA** | **Signatures** |
| 1 | L1F16BSCS0502 | ZAID BIN MOAZ | 2.68 |  |
| 2 | L1F16BSCS0294 | KHIZER HAMEED | 2.83 |  |
| 3 | L1F16BSCS0241 | SHAHZAD MIRAJ | 3.50 |  |

**Advisor’s Consent**

I Prof./Dr./Mr./Ms. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ am willing to guide these students in all phases of above-mentioned project as advisor. I have carefully seen the Title and description of the project and believe that it is of an appropriate difficulty level for the number of students named above.

|  |  |  |
| --- | --- | --- |
| **Note:**  Advisor can’t be changed without prior permission of the Manager Projects and the duration for completion of the Project is 2 regular semesters (approx.) from the date of Registration of Research Project. | Signatures and Date  |  | | --- | |  |   **Advisor** |

**EVALUATOR/REFEREE 1**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| I have carefully read the project proposal and feel that the proposed project is a useful one and of a sufficient difficulty level to justify 2 regular semesters workload for above mentioned students. I have made recommendations in the evaluation form to improve the scope and quality of the project. | | | | | |
|  | | | | Signatures and Date | |
|  |  |  |  |  |  |
|  | | | |  |

**EVALUATOR/REFEREE 2**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| I have carefully read the project proposal and feel that the proposed project is a useful one and of a sufficient difficulty level to justify 2 regular semesters workload for above mentioned students. I have made recommendations in the evaluation form to improve the scope and quality of the project. | | | | | |
|  | | | | Signatures and Date | |
|  |  |  |  |  |  |
|  | | | |  |

**Abstract / Executive Summary**

There is a big difference between workout in trainer’s supervision and workout using a workout application. The reason is that the trainer tells performer about its flaws and the performer can make corrections accordingly. While using a workout application, there is no one to tell you about your flaws and flaws in technique can lead to injuries. Our application will provide supervision for the users doing workout without trainer.

**Introduction and Background**

In existing workout applications, the exercise is explained using description or video which only fulfil the purpose of guide book for the user. These applications do not provide supervision for the user. Supervision is very important for exercise because anyone performing exercise using only a guide book cannot predict its own flaws and these flaws can lead to severe injuries like muscular injury, joints- dislocation, wrist sprain. For this, we need an interactive application which provides supervision by detecting flaws in real-time. To tackle the present problem we proposed a methodology, the user performs exercise in front of the laptop camera facing the screen so that its movement can be seen. The video being recorded in real-time is taken as input and from the video body parts are detected. The detected body parts are then compared to correct data set in the application to find error.

**Statement of the Problem**

The application provides supervision for exercise in real-time to predict flaws.

**Objective(s) / Aim(s) / Target(s)**

As we have described above that the purpose of our proposed methodology is to provide supervision for exercise performed by the user .Thus we can cover those exercises in which body is properly detected(exercises in standing position) in the video. We are doing this for two exercises i.e. **biceps curls, standing barbell curls** etc. The objectives of our workout application are following

* It provides supervision to the user on real-time.
* It predicts the flaws in the exercises, so that to prevent any severe injuries (like muscular injury, joints-dislocation, wrist sprain).

**Completeness Criteria**

The application should detect the body parts, so that error in body postures could be estimated.

**Challenges**

The challenges we have in doing this project are:

1)Processing video on real-time for input in application.

2)To understand working of "open cv(computer vision)" algorithm which is to be used in image processing for detection of body.

3)Making a correct data-set for exercise as data set for body postures related to exercise is not available.

**Knowledge Areas Required**

From our BSCS degree, the areas covered in our project are:

* Advance image processing
* Machine Learning

**Learning Outcomes**

* Advance image processing
* Desktop application development
* Machine learning and its algorithms

**Nature of the End Product / Research Outcomes**

The end product will be an interactive workout application for desktops/laptops which will provide supervision.It can also be used as a software package for application where we need to teach some technique involving body postures.

**Related Work / Literature Survey / Literature Review**

Some of the famous existing workout applications are following

* JEFIT
* Cardio
* Homeworkout

These applications provide good guide for exercises but none of the above provide supervision. Moreover, the method we are using for supervision is used in games specifically the VR games involving devices like **Kinect** etc.

**Deliverables / Work Breakdown Structure**

* **Deliverables:**An application which will provide supervision for workout.
* **Work Breakdown Structure**

**Phase 1:** Making the Requirements Documentation

**Step 1.1:** Writing the Introduction and Background

**Activity 1.1.1:** Writing the Problem Statement.

**Activity 1.1.2:** Writing the Background.

**Activity 1.1.3:** Writing the Scope.

**Activity 1.1.4:** Writing the Objective(s)/Aim(s)/Target(s).

**Activity 1.1.5:** Writing the Challenges.

**Activity 1.1.6:** Writing the Learning Outcomes.

**Activity 1.1.7:** Writing the End Product.

**Activity 1.1.8:** Writing the Completeness Criteria.

**Activity 1.1.9:** Writing the Business Goals.

**Activity 1.1.10:** Writing the Literature Survey/Review.

**Step 1.2:** Writing the Overall Description

**Activity 1.2.1:** Writing the Product Features.

**Activity 1.2.2:** Defining the User Classes and Characteristics.

**Activity 1.2.3:** Define the Operating Environment.

**Activity 1.2.4:** Denote the Design and Implementation Constraints.

**Activity 1.2.5:** Writing down the Assumptions and Dependencies.

**Step 1.3:** Mention the Functional Requirements

**Activity 1.3.1:** Draw the System Architecture Diagram

**Activity 1.3.2:** Write down all the possible Use Cases.

**Activity 1.3.3:** Draw all the use case diagrams, sequence diagram and abstract class diagram.

**Step 1.4:** Mention the Non-Functional Requirements

**Activity 1.4.1:** Mention the performance requirements.

**Activity 1.4.2:** Mention the Security Requirements.

**Activity 1.4.3:** Write down any additional software quality attributes that are left.

**Step 1.5:** Write down any Other Requirements you might have missed.

**Phase 2:** Perform System Decomposition and make Architecture and User

Interface Design

**Step 2.1:** Do System Decomposition

**Activity 2.1.1:** Perform functional (modular) decomposition.

**Activity 2.1.2:** Perform Object Oriented decomposition.

**Activity 2.1.3:** Perform Feature Oriented decomposition.

**Step 2.2:** Develop complete Architecture of application

**Activity 2.2.1:** Select a suitable Architecture styles.

**Activity 2.2.2:** Arrange the aforementioned modules according to it.

**Activity 2.2.3:** Mention the architecture styles used and the reason behind using them.

**Activity 2.2.4:** Label architecture document to map it in accordance with feature decomposition.

**Step 2.3:** Design the User Interface.

**Phase 3:** Make the Detailed Design Document

**Step 3.1:** Draw the complete Class Diagram.

**Step 3.2:** Draw the complete ER Diagram.

**Step 3.3:** Mention the information on the use of design patterns while designing the modules.

**Phase 4:** Coding the Application.

**Step 4.1:** Breaking the problem down into smaller parts.

**Step 4.2:** Coding each small part.

**Step 4.3:** Connecting all the small parts to each other.

**Phase 5:** Testing the Application.

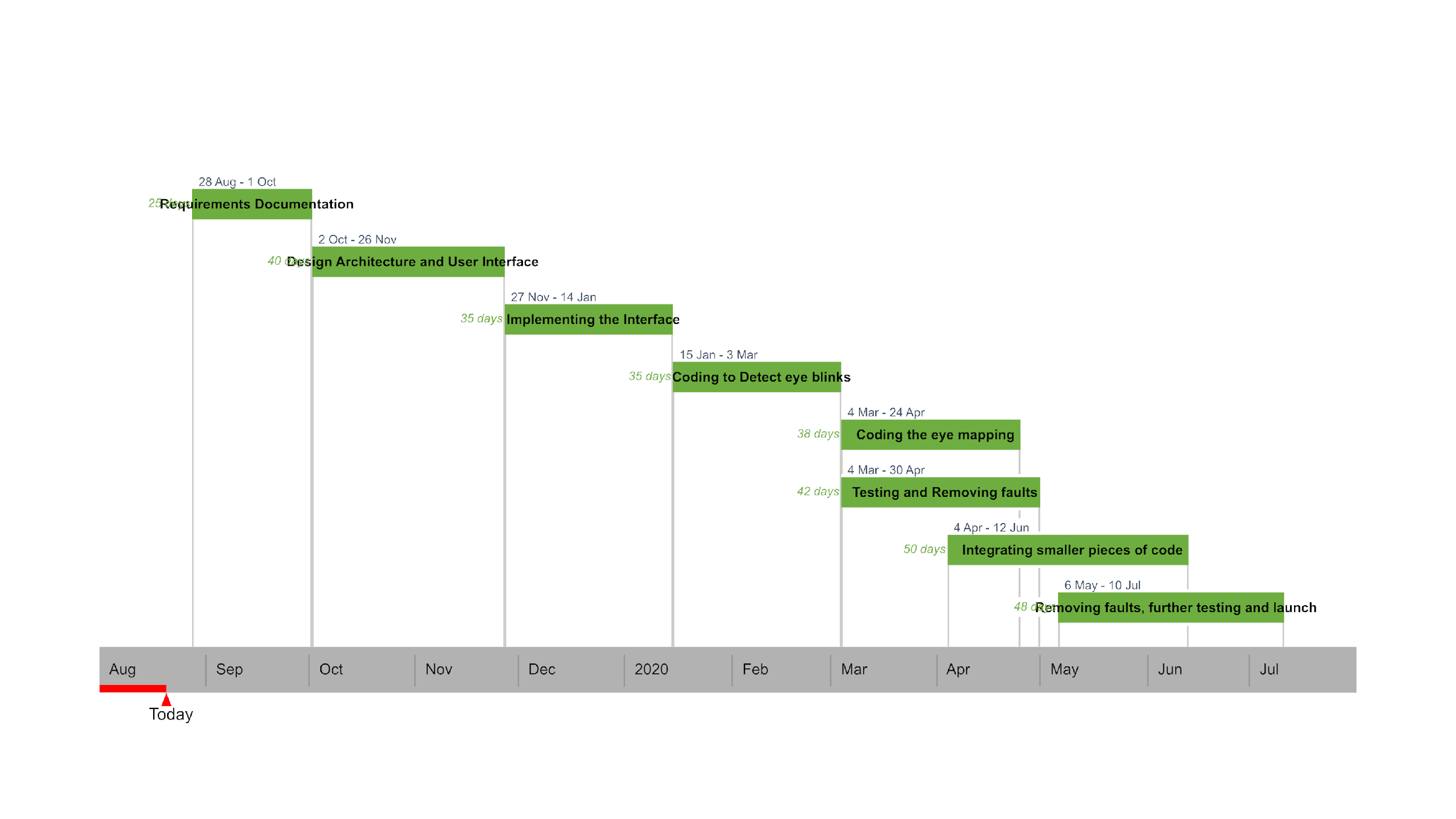
**Step 5.1:** Decomposing each small unit of the code.

**Step 5.2:** Testing each unit of code individually.

**Step 5.3:** Combine the code now.

**Step 5.4:** Test all the code collectively now.

**Project Plan / Project Schedule / Project Timetable / Project Calendar**

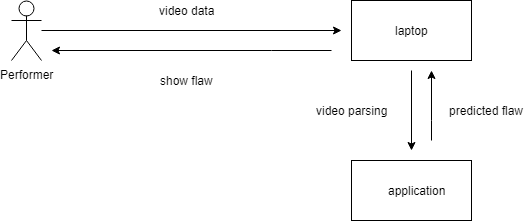


**Resources Required**

The application requires a laptop or desktop with a web-cam.

**Miscellaneous**

**Sketch of Proposed Solution (For Research-based and Hardware-Oriented Projects Only)**



**References/Bibliography**

International Symposium on Computer Network and Multimedia Technology.URL<https://ieeexplore.ieee.org/document/5374792>

Body detection algorithms

URL<https://github.com/opencv/opencv/blob/master/samples/dnn/object_detection.py>