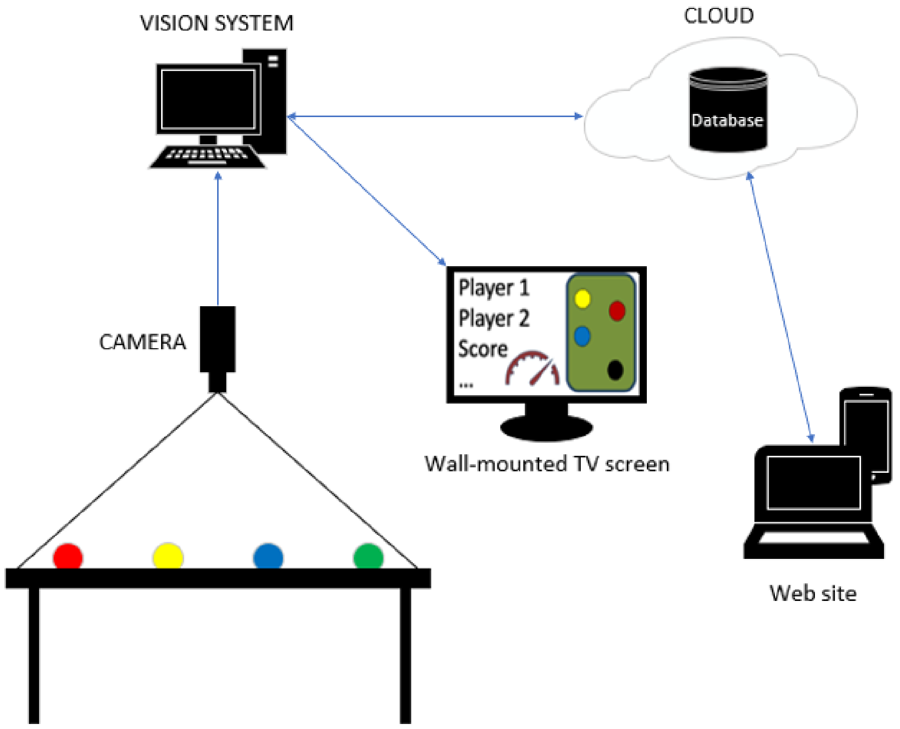
University of
South-Eastern Norway

www.usn.no

PRH612 Bacheloroppgave

Implementation of Cloud-based Vision System with focus on Internet of Things, Web Technology, Machine Learning and Cyber Security



IA6-1-20

The University of South-Eastern Norway takes no responsibility for the results and conclusions in this student report.

University of
South-Eastern Norway

www.usn.no

Course: PRH612 Bacheloroppgave

Title: Implementation of Cloud-based Vision System with focus on Internet of Things, Web Technology, Machine Learning and Cyber Security

This report forms part of the basis for assessing the student's performance in the course.

Project group: IA6-1-20 **Availability:** Open

Group participants: Kevin Skogstad Kristiansen
Joakim Thommesen Larsson
Henrik Andreas Rustad

Supervisor: Maths Halstensen

Hans-Petter Halvorsen

Robert Immerstein

Project partner: **Grenland Biljardklubb**

Approved for archiving: ____
(supervisor signature)

The University of South-Eastern Norway takes no responsibility for the results and conclusions in this student report.

Summary:

Due to an ever-modernizing society, most arcade and pub-like games have yet to modernize and adapt to the modern world. The world today is based on accessibility and shareability, but due to the physical and local aspects of many games, such as pool and darts, their popularity is hindered. Therefore, in cooperation with Grenland Biljardklubb, an interest to establish a digital foundation for what the modern pool game can become has surfaced.

The goal of the project was to create a system capable of presenting the pool games played. To achieve this goal, a Vision System was created that acquire image data of a pool game and returns the game state data. The Vision System is the judge for the games being played, accounting for balls being shot down, and which player plays which type of balls. The games played are presented locally through the Vision System and online with a website created during the project. The website functions as a mediator between the user and the system. The website reads data from the Vision System, displays data, takes in user input, and makes it possible to interact with the system.

The University of South-Eastern Norway takes no responsibility for the results and conclusions in this student report.

Preface

This report is the result of a 6th semester bachelor thesis completed by students at the University of South-Eastern Norway (USN). The bachelor group studies Computer Science and Industrial Automation at the Faculty of Technology, Natural Sciences and Maritime Sciences at USN.

The group would like to thank our supervisor Hans-Petter Halvorsen for his contributions by use of his extensive lecture notes and personal technical blog which has been found to be very useful.

The cover photo used, was created by the group to give an overview of the solution given in the project.

The applications and source code are stored on an external flash drive.

A link to the project website: https://billiardbachelor.azurewebsites.net/

During the project, an array of software has been used to make development easier:

- erwin Data Modeler r9.8
- LabVIEW 2019
- MS Excel 2019
- MS PowerPoint 2019
- MS Project 2019
- MS Visio 2019
- MS Word 2019
- Visual Studio 2019

Readers with foreknowledge regarding LabVIEW, ASP.Net Core 3.1, and Scripted Query Language (SQL) will have an easier time understanding some of the concepts in this report.

Porsgrunn, 18.05.20

Nomenclature

DIN	Deutsches Institut für Normung
DNS	Domain Name System
ER	Entity Relationship
GDPR	General Data Protection Regulation
IP	Internet Protocol
ISP	Internet Service Provider
MMR	Match Making Ranking
NUC	Next Unit of Computing
RGB	Red Green Blue
ROI	Region of Interest
SQL	Scripted Query Language
USN	University of South-Eastern Norway

Contents

1 .. Introduction 7

2 .. System Description 8

2.1 Hardware 9

3 .. Vision System 11

3.1 Application Development 11
3.2 Image Processing 11
3.3 Object Detection 14
3.4 Object Classification 15
3.5 Digital Judge 16

4 .. Database 18

4.1 Structure 18

5 .. Website 21

5.1 Use Case 21
5.2 Web pages 21
5.3 Password Hashing 26

6 .. Discussion 27

6.1 Vision System 27
6.2 Website 28
6.3 Future Development 29

7 .. Summary 31

8 .. References 32

Appendices 33

Appendix A 34
Appendix B 37
Appendix C 38
Appendix D 43
Appendix E 48
Appendix F 49

1 Introduction

Due to an ever-modernizing society, most arcade and pub-like games have yet to modernize and adapt to the modern world. The world today is based on accessibility and shareability, but due to the physical and local aspects of many games, such as pool and darts, their popularity is hindered. Therefore, in cooperation with Grenland Biljardklubb, an interest to establish a digital foundation for what the modern pool game can become has surfaced.

The scope of this project involves detection and classification of pool balls on a pool table by use of a computer. A digital judge will determine which player shoots, which balls have been scored, and who wins, and loses, the games.

Information from the detection and classification will be sent to a cloud-based database and used on a website created in this project. Users on the website should be able to create an account, start new games with other users, and receive a digital representation of ongoing games in the system through the website.

The goal of this project is to create a functional prototype which handles pool games digitally. An assessment on the requirements of this project, and the creation of the prototype, will be done to determine the best approach towards the task at hand.

To achieve the goal put forth in this project, LabVIEW is utilized to create a Vision System. The LabVIEW Vision System detects and classifies pool balls and provides data to the database. A website and a cloud-based database is also created. The website reads from, and writes to, the database, and provides the user with an access point to the functionality provided by this project, in a user-friendly fashion. The database is created on the Azure platform and used to store all necessary data on the players and games.

The following list will cover the report structure:

- Chapter 2 gives a general description of the system. This encompasses hardware, software, goals, and the purposes of the different aspects of the system.
- Chapter 3 provides a detailed description of the inner workings of the LabVIEW Vision System.
- Chapter 4 details the database, its data structure, and the data stored.
- Chapter 5 encompasses the website and its functionality.
- Chapter 6 covers the result of the undertaking.
- Chapter 7 addresses on the group discussion on the project.
- Chapter 8 contains all the references used in the report.

2 System Description

A Vision System is created to detect and classify pool balls at a pool table. The Vision System keeps track of the score on a current game and transfers the location data of every pool ball on the table to a cloud-based database. The application runs on a Next Unit of Computing (NUC) with connection to the internet. A graphical interface is created to display an ongoing game and an updated scoreboard on a wall-mounted TV screen in the pool hall.

The system uses Azure Databases to store the data utilized throughout the system. The Azure Database stores data on the players, and the games played, using this system. Each game played is stored from its inception, with all the rounds of each game stored as well. This allows for a replay function, as well as highlighting moves.

A website is created to establish a shared platform where games can be displayed digitally, as well as gaining the possibility to create games, tournaments, and challenging other players. The website will be the starting point of each game and tournament, as well as the streaming platform for current games.

In Figure 2-1 an overview of the system is shown.

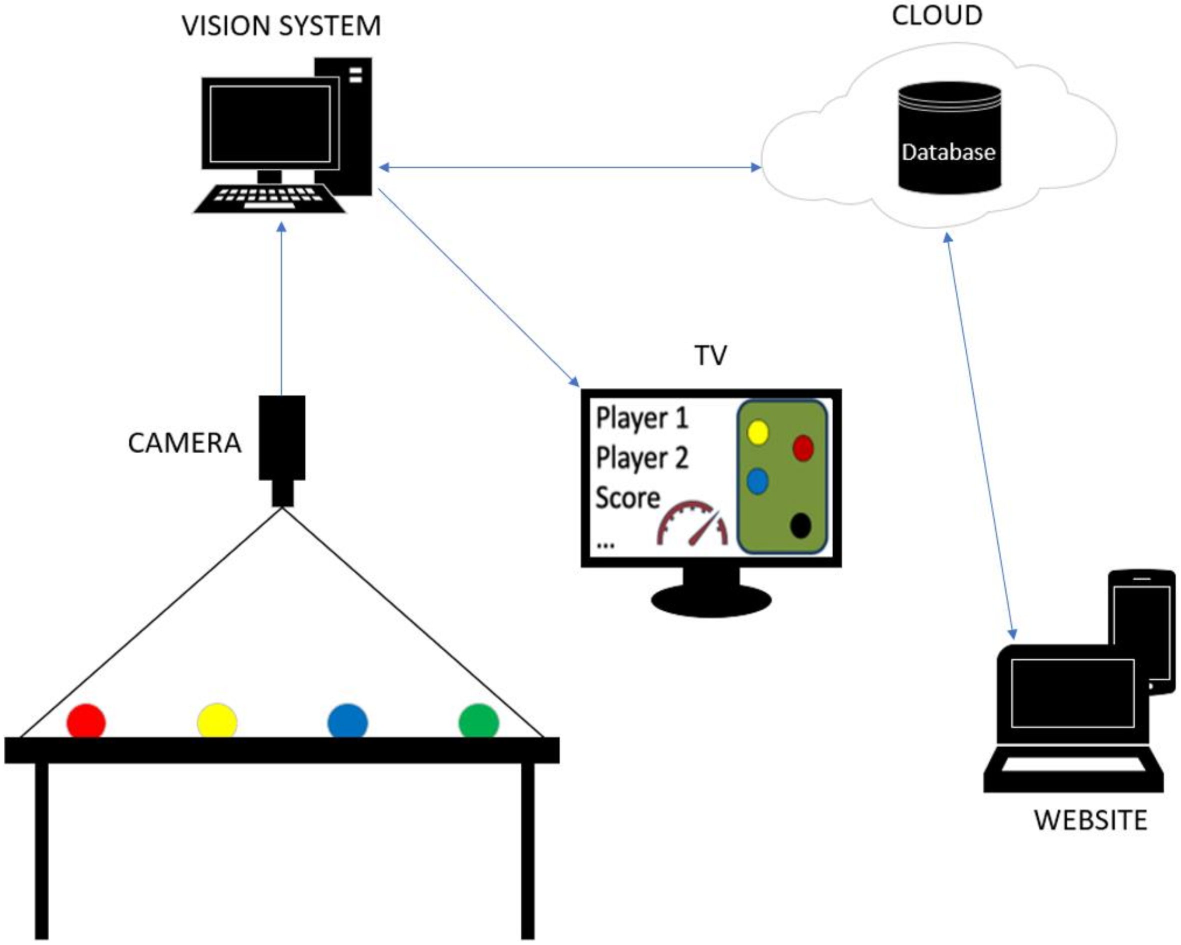


Figure 2-1: Graphical System Description

2.1 Hardware

The system requires several hardware components to function. Different cameras and computers have been tested to find the best approach for the system. The next subchapters will describe the hardware the system uses.

2.1.1 Basler Camera

For the Vision System, a Basler acA5472-17uc camera has been used for recording, and snapshotting, images from the game. The Basler camera has high resolution and many advanced settings, which makes it possible to achieve high accuracy, and precision, for image processing. For details about settings for the Basler camera, see Appendix B.

2.1.2 Camera Mount

A ceiling mounting has been created and installed to give the Vision System the best possible view and mobility. The mounting makes it easy to adjust the angle, vertical, and horizontal position. The ceiling consists of a milled Deutsches Institut für Normung (DIN) rail to adjust the horizontal position, a ¼" joint at both ends to adjust the vertical position, and a joint allowing for angle adjustment. In Figure 2-2, an illustration of the camera mount is shown.

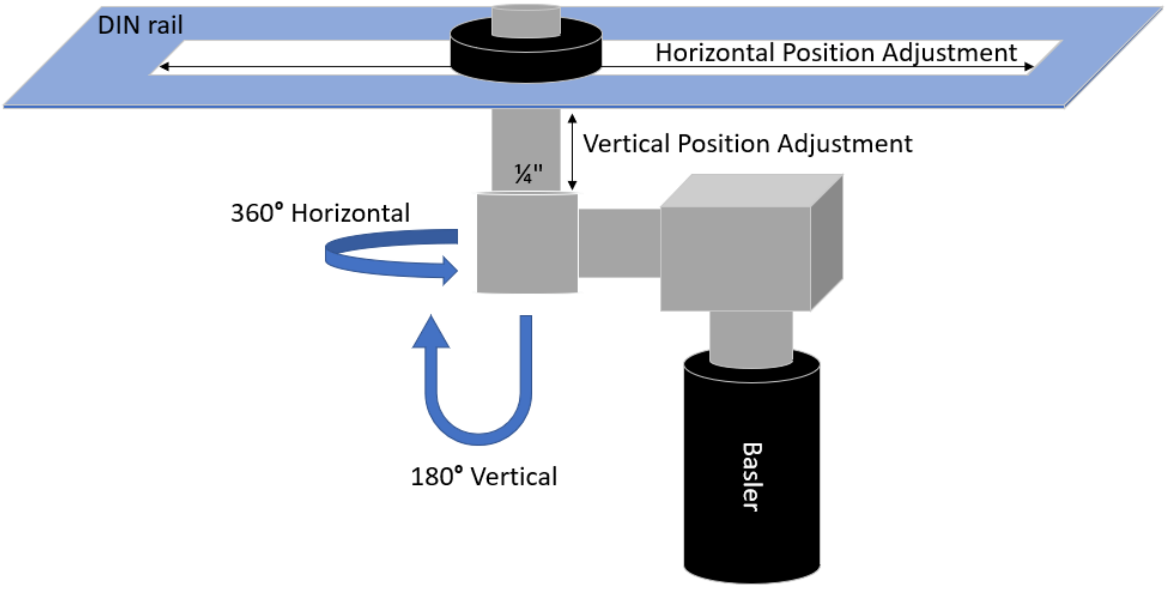


Figure 2-2: Camera Mount

2.1.3 Next Unit of Computing

A NUC is used as the computer for this system. A NUC is a high-performance small form factor computer designed by Intel [1]. In this system, the NUC is responsible to acquire the image data from the Basler camera, run the Vision System application, display the application to a television, and have connection to the cloud-based database. In Figure 2-3, an illustration of the NUC interactions is shown.

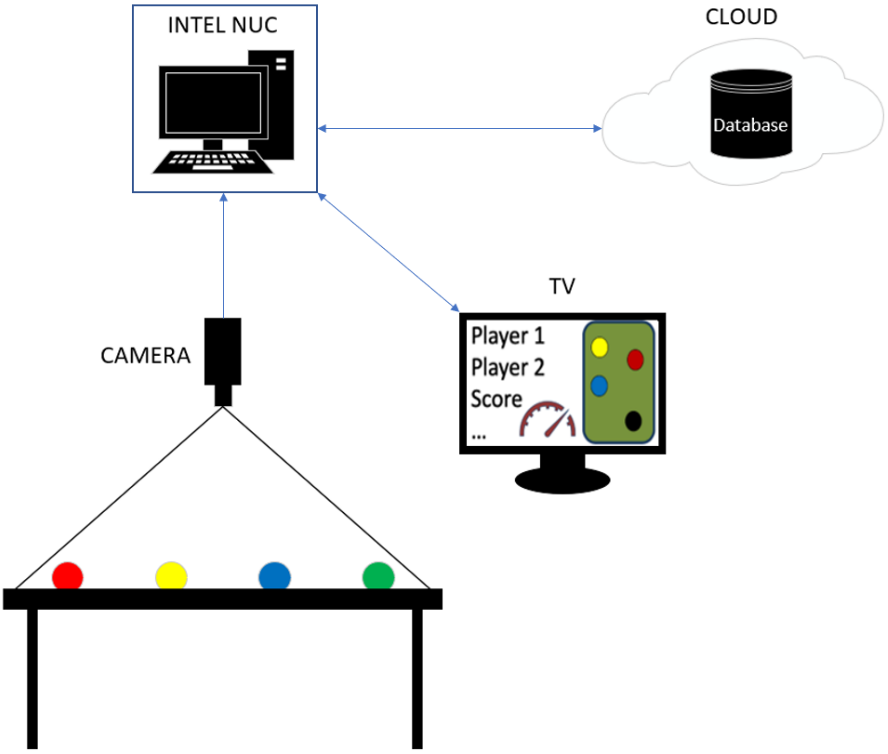


Figure 2-3: NUC Interactions

3 Vision System

The Vision System is responsible to detect, classify the pool balls, and transfer the location to a cloud-based database. The Vision System is also responsible to keep track of the score on an ongoing game and decide which player to shoot. A live video of the game is transferred from the Vision System to a wall-mounted TV in the pool hall with an updated scoreboard. For this Vision System, the camera is used for image and video acquisition. The application is developed in LabVIEW and it is running on the NUC, which is connected to the Azure Database. In Figure 3-1, an overview of the Vision System is shown.

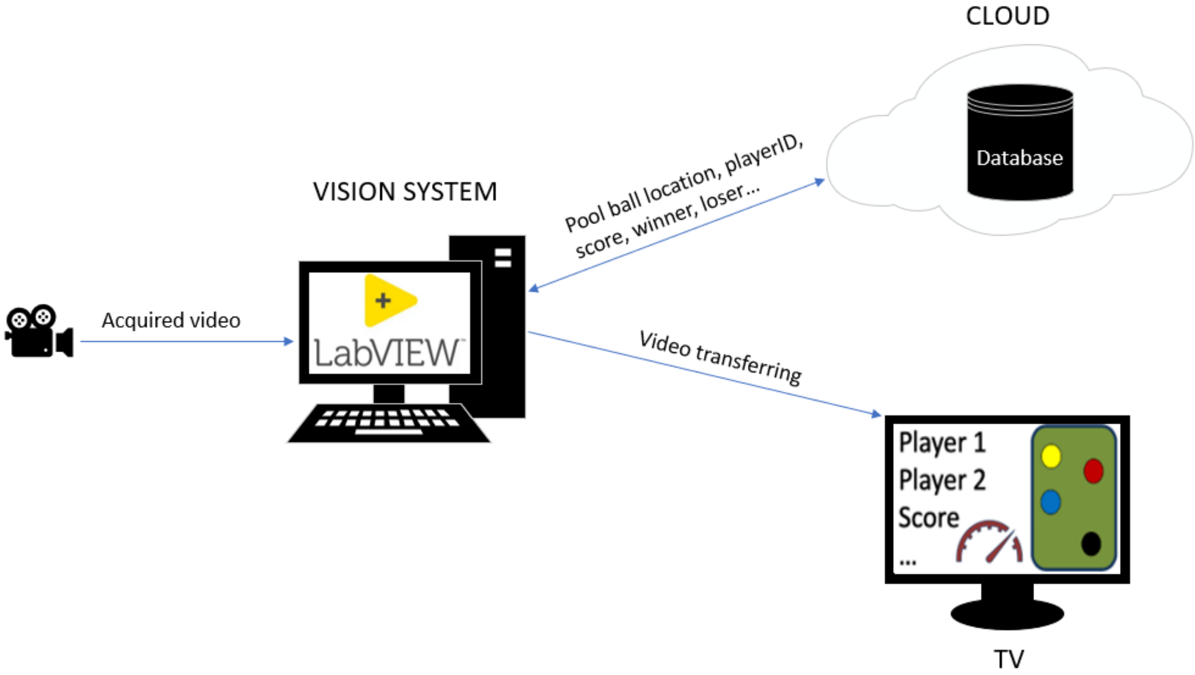


Figure 3-1: Vision System Overview

3.1 Application Development

The Vision System requires high performance in image processing and data transferring, therefore, different software, to develop the application, was tested. Both LabVIEW and C# were explored to find the best approach for this system. LabVIEW has several integrated image processing tools that make it easier to process images compared to C#. Since the image processing tools in LabVIEW has high performance, accuracy, and ease of use compared to C#, LabVIEW was chosen for this application. The graphical user interface (GUI), and a user guide for the application, is represented in Appendix C.

3.2 Image Processing

The purpose of processing the incoming images from the camera is to extract the pool balls from the background. To accomplish this task, an integrated image processing tool, i.e. a

Vision Assistant in LabVIEW, has been used. An image from the ongoing game is sent to the Vision Assistant, which runs several functions in a sequence, to extract the pool balls from the background. Since the pool balls has different colors, there are four Vision Assistants that runs in series to extract every pool ball on the table. In Figure 3-2, an input and output image from a Vision Assistant is shown. The details, on the workflow, of one of the Vision Assistants will be described, in order, in the coming subchapters. For details on every image processing sequence, see Appendix D.

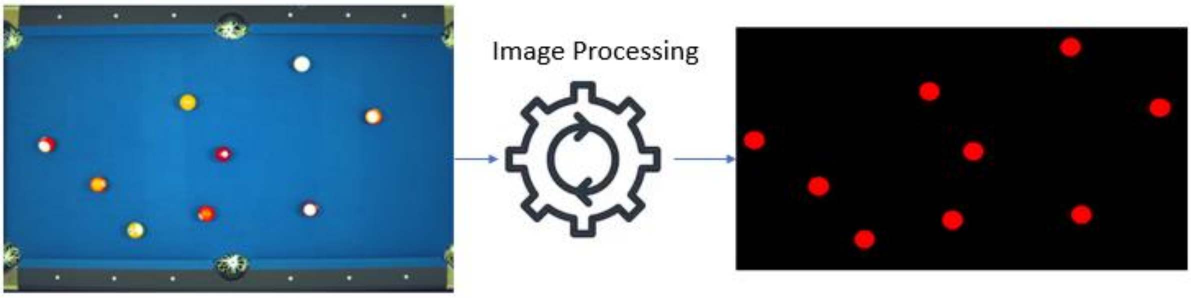


Figure 3-2: Image Processing Input and Output

3.2.1 Image Mask

The first step of the sequence is to select a region of interest (ROI). The ROI, in the incoming image, is the area that the ball can appear in during a game, i.e. the table matt. By removing the unwanted parts of the image, the processing time, and the probability of errors are reduced.

3.2.2 Color Operator and Brightness

The blue pool cloth covers a lot of the image background and is a prime target for removal. This is done using a color operator in the Vision Assistant. The color to subtract is chosen using a color histogram generated by the Vision Assistant.

After subtracting the background, the Vision Assistant then adjusts the brightness, contrast, and gamma to enhance the difference between the pool balls and the background even more.

3.2.3 Color Plane Extraction

The incoming image is a 32-bit Red, Green, Blue (RGB) image where the red, green, and blue colors have 8-bits to represent the scale of each color in a pixel [2]. A color plane extraction isolates an 8-bit color plane, i.e. isolating either the red, green, or blue colors, from the 32-bit image. The result is the intensity of the chosen color in the image. In Figure 3-3, a red color plane extraction is used to extract the red color from the image. The extracted image is an 8-bit image, where every pixel has a value from 0-255 that represent the red value. The brighter sections of the image have a higher red intensity value, while the darker sections have low, or no, red value.

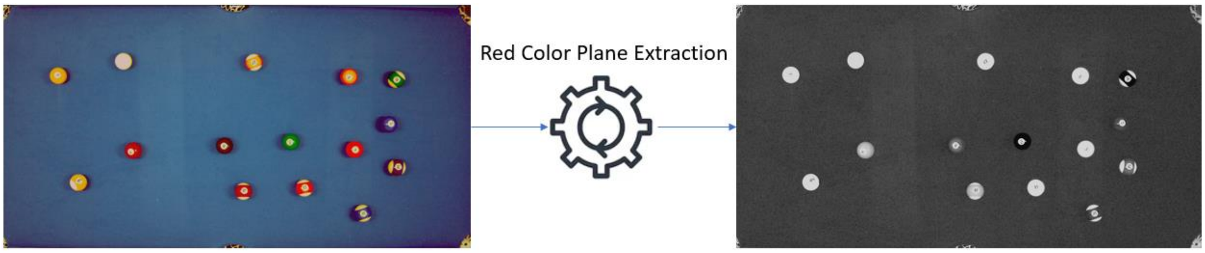


Figure 3-3: Red Color Plane Extraction

3.2.4 Threshold

This step converts the extracted 8-bit image, described in 3.2.3, into a single bit image. This is done using a threshold. The purpose of the threshold is to determine whether a pixel is a part of a pool ball or the background. Figure 3-4 shows the result of applying a threshold in one of the Vision Assistants. The threshold value, in this Vision Assistant, is set to 83, which implies that every pixel that has a higher intensity value than 83, is set to 1, i.e. here shown in red.

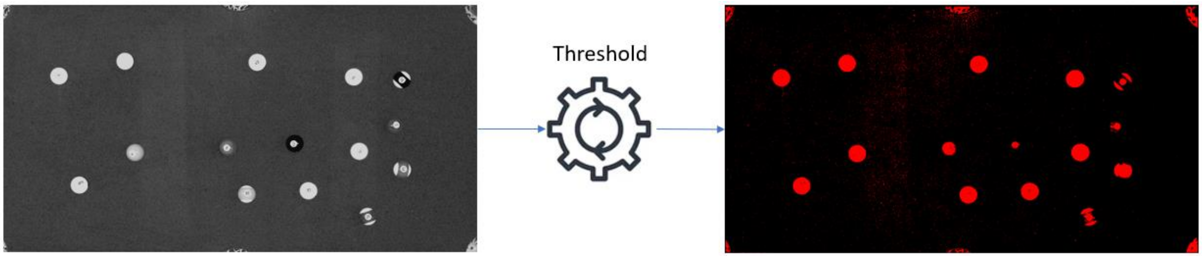


Figure 3-4: Threshold

3.2.5 Advanced Morphology

To reduce unwanted noise from the single bit image, several advanced morphology steps are used in sequence. First, a function to remove the border objects is used. Removing the border objects clears the sockets displayed along the image border. Then a function to remove small objects is used. After the border objects and the small objects are removed, a convex hull function is used to fill the gaps in the pool balls. Figure 3-5 shows the difference in the binary image, before and after the morphology sequence. In the output image, multiple balls have been lost in the advanced morphology process. In return, what is left are very precise readings. Due to this loss in pool balls, multiple Vision Assistants are needed to cover every ball on the table.

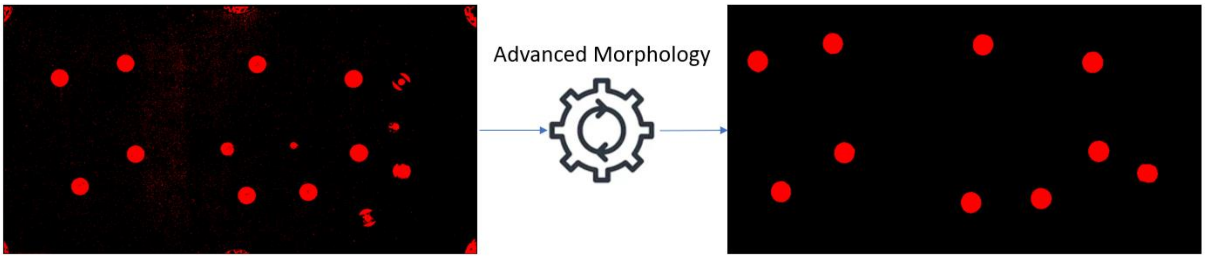
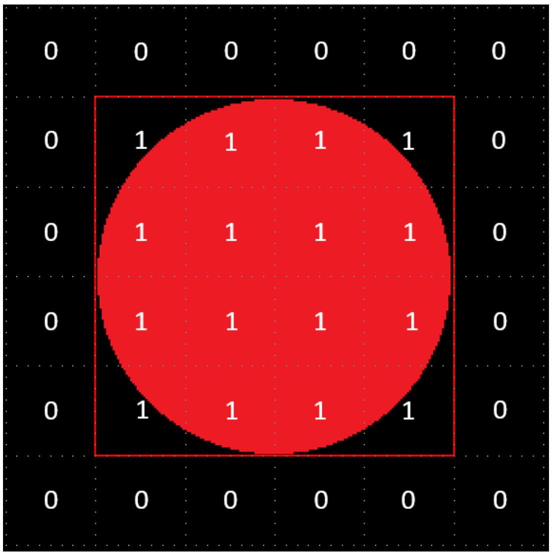


Figure 3-5: Advanced Morphology

3.3 Object Detection

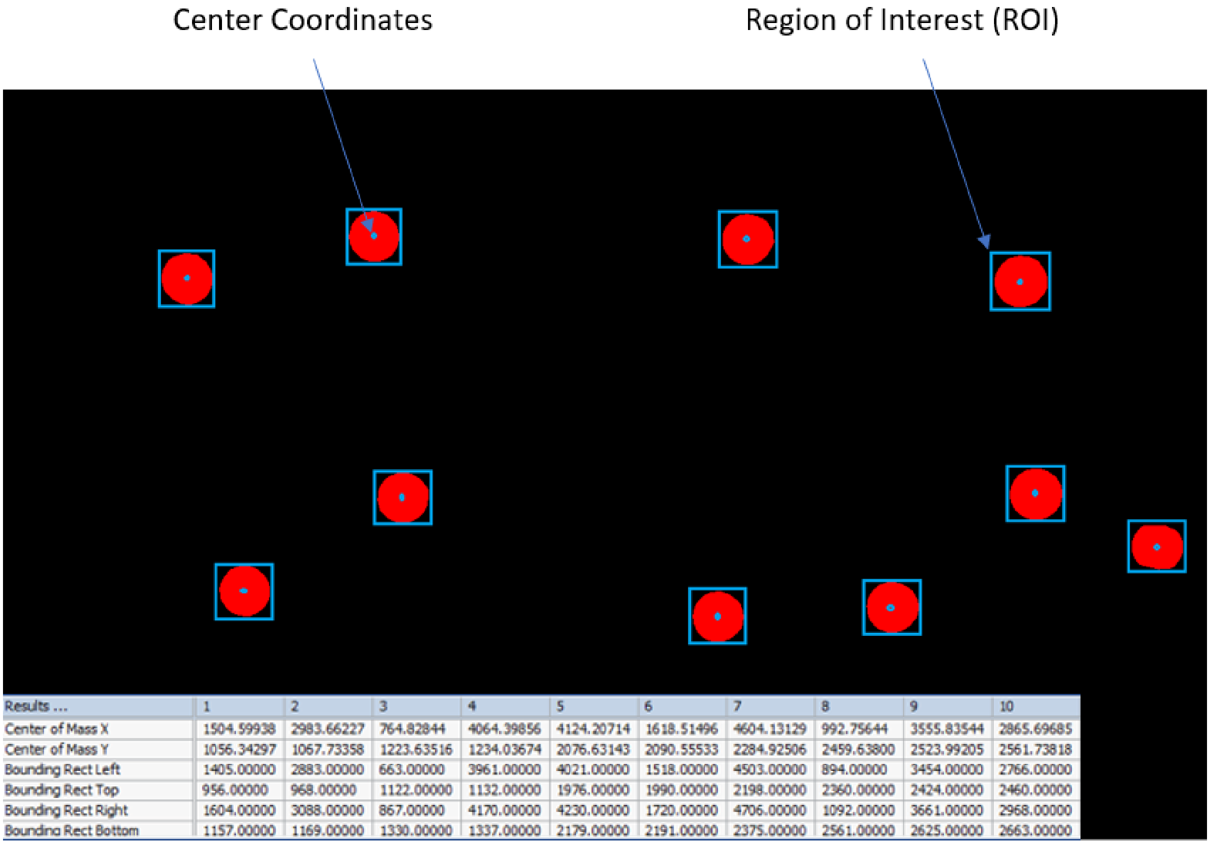
The binary image from the image processing has extracted the pool balls from the background. To classify the pool balls as an object, a particle analysis function is used in the Vision Assistant tool. The particle analysis function uses an algorithm to find patterns in the binary image. Figure 3-6 shows a simplified illustration of how the particle analysis algorithm classifies a pool ball as an object.



0	0	0	0	0	0
0	1	1	1	1	0
0	1	1	1	1	0
0	1	1	1	1	0
0	1	1	1	1	0
0	0	0	0	0	0

Figure 3-6: Object Detection

The particle analysis function stores a lot of information about every pool ball, such as the bounding rectangle, i.e. area, coordinates, and center coordinates. The center coordinates are used to store the position of the pool balls in the cloud-based database. The bounding rectangle coordinates are used to create a ROI for every pool ball on the table. It is crucial that the ROI is accurate, because this is used to classify the pool balls after the detection. The use of ROI is described in next subchapter. In Figure 3-7 the center coordinates, ROI and the output array are shown.



Results ...	1	2	3	4	5	6	7	8	9	10
Center of Mass X	1504.59938	2983.66227	764.82844	4064.39856	4124.20714	1618.51496	4604.13129	992.75644	3555.83544	2865.69685
Center of Mass Y	1056.34297	1067.73358	1223.63516	1234.03674	2076.63143	2090.55533	2284.92506	2459.63800	2523.99205	2561.73818
Bounding Rect Left	1405.00000	2883.00000	663.00000	3961.00000	4021.00000	1518.00000	4503.00000	894.00000	3454.00000	2766.00000
Bounding Rect Top	956.00000	968.00000	1122.00000	1132.00000	1976.00000	1990.00000	2198.00000	2360.00000	2424.00000	2460.00000
Bounding Rect Right	1604.00000	3088.00000	867.00000	4170.00000	4230.00000	1720.00000	4706.00000	1092.00000	3661.00000	2968.00000
Bounding Rect Bottom	1157.00000	1169.00000	1330.00000	1337.00000	2179.00000	2191.00000	2375.00000	2561.00000	2625.00000	2663.00000

Figure 3-7: Object Detection Information

3.4 Object Classification

To classify the pool balls, a color spectrum has been used in the ROI from the object detection. The color spectrum calculates the percentage of every color in the ROI for each pool ball. In Figure 3-8, the ROI of every pool ball is placed into the raw image, where the color spectrum does the calculations. A color matrix is created to find the correlations between the color spectrum calculations and the pool balls. To obtain a precise classification, the color matrix is based on several different images, from ten different games played. The color matrix is shown in Figure 3-9.

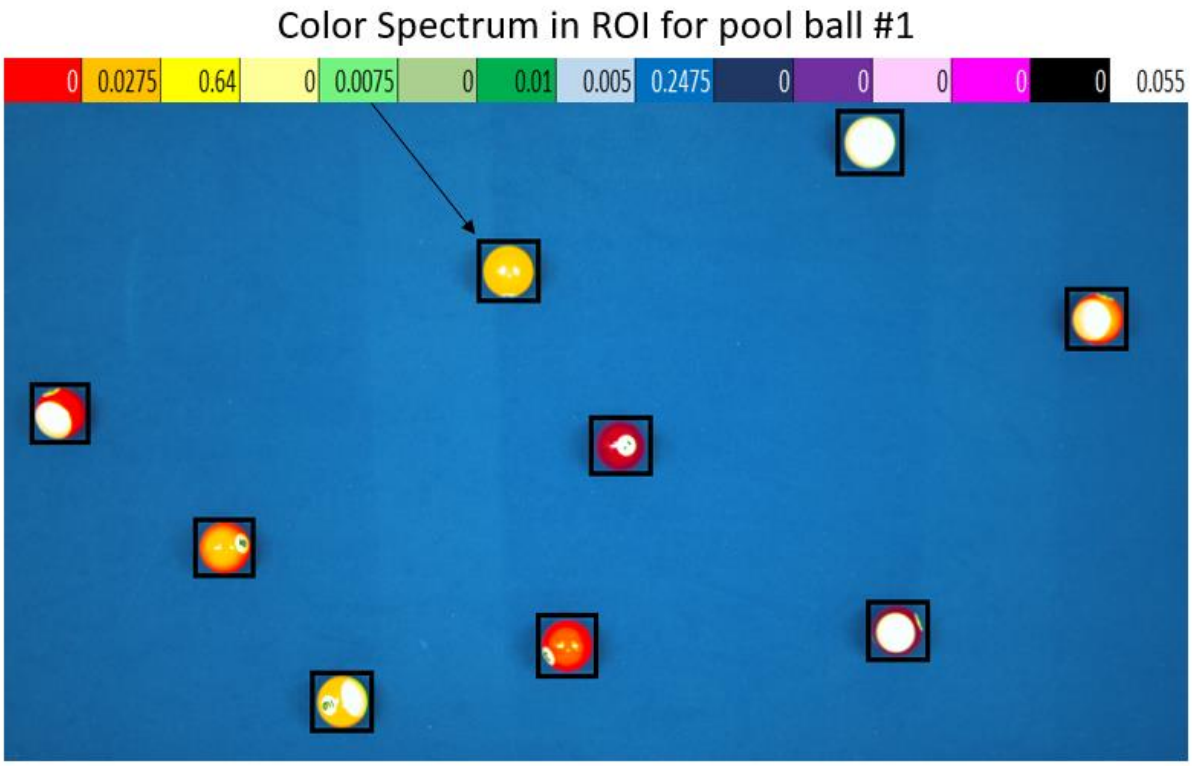
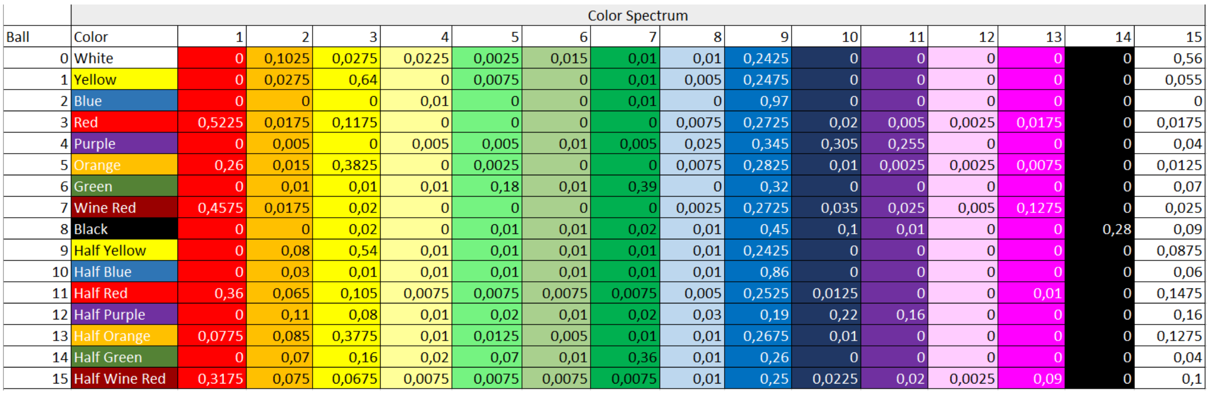


Figure 3-8: Color Spectrum In ROI



		Color Spectrum														
Ball	Color	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	White	0	0,1025	0,0275	0,0225	0,0025	0,015	0,01	0,01	0,2425	0	0	0	0	0	0,56
1	Yellow	0	0,0275	0,64	0	0,0075	0	0,01	0,005	0,2475	0	0	0	0	0	0,055
2	Blue	0	0	0	0,01	0	0	0,01	0	0,97	0	0	0	0	0	0
3	Red	0,5225	0,0175	0,1175	0	0	0	0	0,0075	0,2725	0,02	0,005	0,0025	0,0175	0	0,0175
4	Purple	0	0,005	0	0,005	0,005	0,01	0,005	0,025	0,345	0,305	0,255	0	0	0	0,04
5	Orange	0,26	0,015	0,3825	0	0,0025	0	0	0,0075	0,2825	0,01	0,0025	0,0025	0,0075	0	0,0125
6	Green	0	0,01	0,01	0,01	0,18	0,01	0,39	0	0,32	0	0	0	0	0	0,07
7	Wine Red	0,4575	0,0175	0,02	0	0	0	0	0,0025	0,2725	0,035	0,025	0,005	0,1275	0	0,025
8	Black	0	0	0,02	0	0,01	0,01	0,02	0,01	0,45	0,1	0,01	0	0	0,28	0,09
9	Half Yellow	0	0,08	0,54	0,01	0,01	0,01	0,01	0,01	0,2425	0	0	0	0	0	0,0875
10	Half Blue	0	0,03	0,01	0,01	0,01	0,01	0,01	0,01	0,86	0	0	0	0	0	0,06
11	Half Red	0,36	0,065	0,105	0,0075	0,0075	0,0075	0,0075	0,005	0,2525	0,0125	0	0	0,01	0	0,1475
12	Half Purple	0	0,11	0,08	0,01	0,02	0,01	0,02	0,03	0,19	0,22	0,16	0	0	0	0,16
13	Half Orange	0,0775	0,085	0,3775	0,01	0,0125	0,005	0,01	0,01	0,2675	0,01	0	0	0	0	0,1275
14	Half Green	0	0,07	0,16	0,02	0,07	0,01	0,36	0,01	0,26	0	0	0	0	0	0,04
15	Half Wine Red	0,3175	0,075	0,0675	0,0075	0,0075	0,0075	0,0075	0,01	0,25	0,0225	0,02	0,0025	0,09	0	0,1

Figure 3-9: Color Matrix

3.5 Digital Judge

A digital judge is implemented to decide which player shoots, which ball is scored, and which player wins, or loses, a game. Before the judge starts, the players must choose who started, and what ball type the players are playing. When a player has shot, and the image processing from the snapshot is done, the digital judge compares the new snapshot to the previous snapshot. Based on a position analysis, the digital judge decides which player should shoot next. If one or more balls are removed between the current and previous snapshots, the scoreboard is updated to display the missing balls. This is done according to the interface shown in Figure 3-10. When the black ball is shot down, the judge decides who wins, and who loses.

All players are ranked according to their Match Making Ranking (MMR), a value which represents the skill of the players. When a game is finished, the players will either gain, or lose, MMR. The gains and losses are calculated depending on the difference in MMR between the players, the win ratio of the winner, and loss ratio of the loser. The MMR change is always positive for the winner and negative for the loser. If the difference in MMR is drastic, the gain and loss of MMR is drastic in return. The formula describing this behavior can be found in Appendix E.

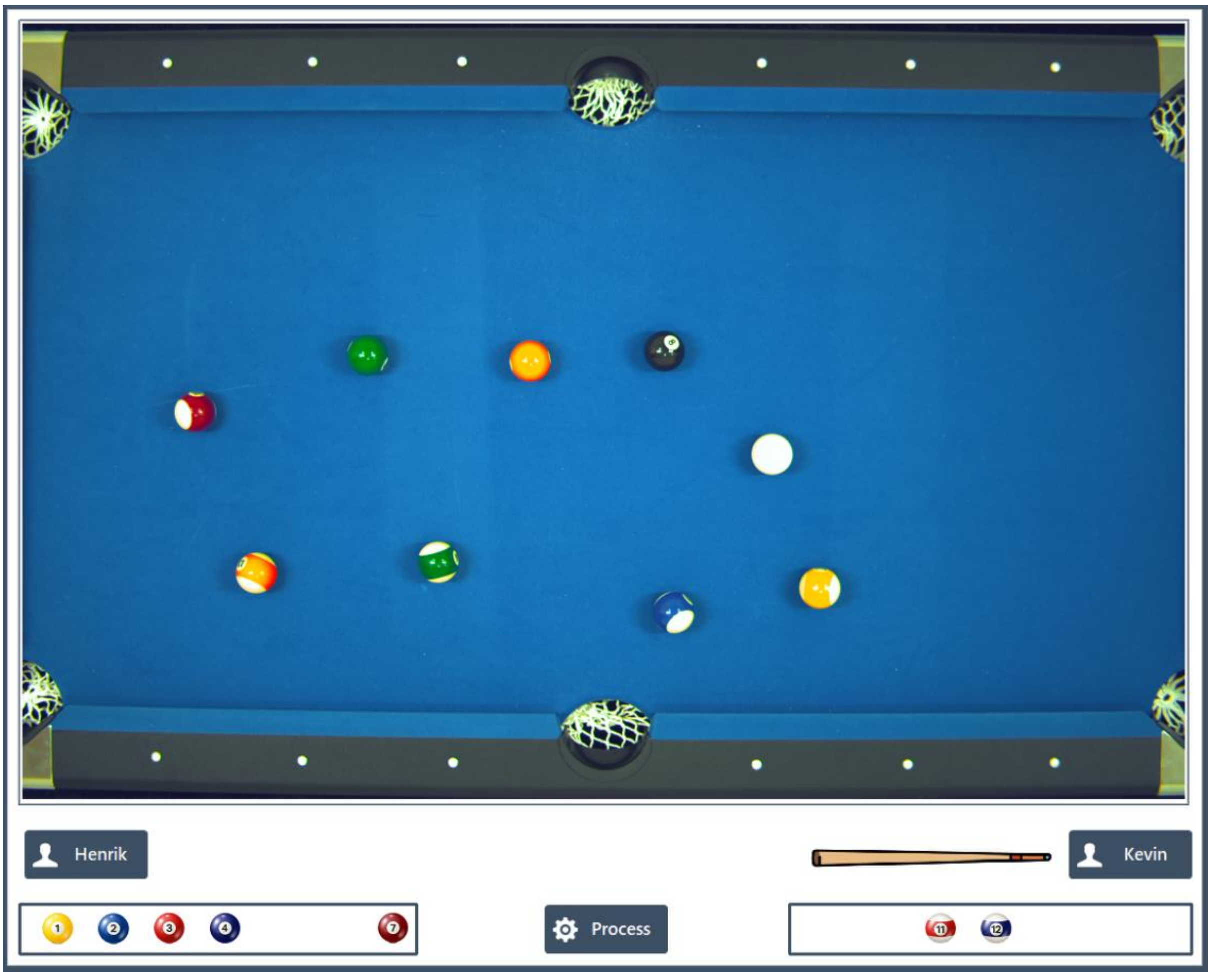


Figure 3-10: In Game Interface

4 Database

The database is used to store data for both the Vision System and the website in this project. The database acts as a buffer between the website and the Vision System. The website utilizes the database to enable user interaction, while the Vision System uses the database for game state tracking. The website accesses the entire database, while the Vision System is mainly focused on the game, balls, and players in games.

Microsoft Azure is utilized to store all the data as an off-site database. This ensures good stability and provides a Domain Name System (DNS) ready address, instead of an Internet Protocol (IP) address which might change with a power outage or Internet Service Provider (ISP) failure.

4.1 Structure

The database contains information regarding games played and the players playing. See Figure 4-1 to see the Entity Relationship (ER) in the database.

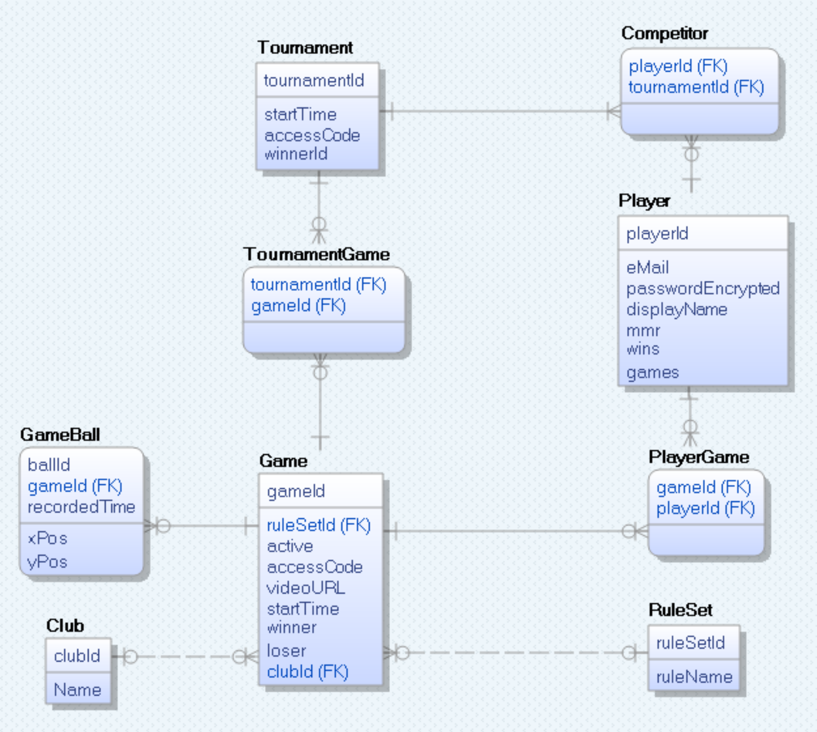
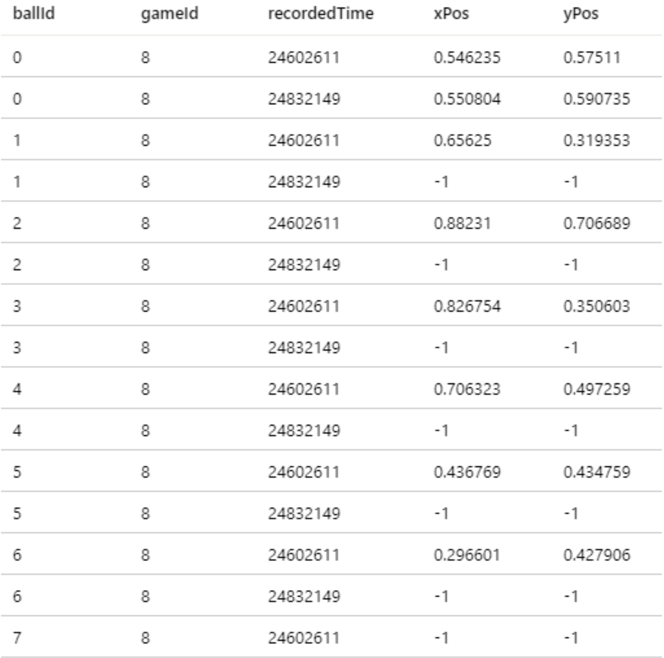


Figure 4-1: The ER Diagram

The “Game” table contains information regarding the state of any game when players are using the service through the Vision System. The “Game” table uses the “Ruleset” and “Club” tables to identify the rules used in the game, and the club the game is being played in respectively.

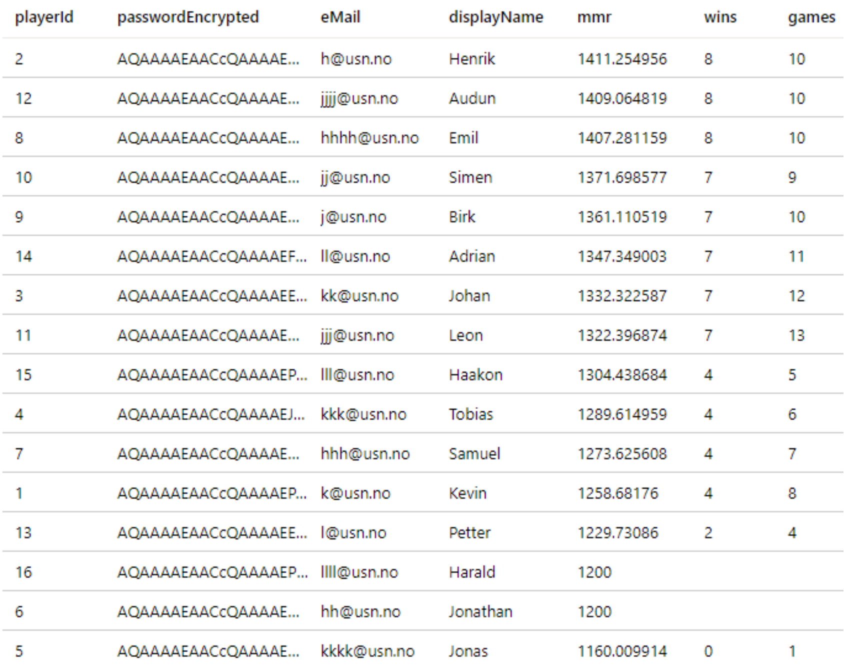
The “GameBall” table contains data regarding the state of the balls on the table during a game. The three primary keys “ballId”, “gameId”, and “recordedTime” is used to identify the balls in such a way that the “ballId” always stays between 0 and 15. 0 being the white ball, while the other “ballId”’s match with the identifying number on each ball. The “xPos” and “yPos” represents the position of the ball at that specific time, with “-1” in either representing a missing ball. In Figure 4-2, the balls of an active game are displayed from the database.



ballId	gameId	recordedTime	xPos	yPos
0	8	24602611	0.546235	0.57511
0	8	24832149	0.550804	0.590735
1	8	24602611	0.65625	0.319353
1	8	24832149	-1	-1
2	8	24602611	0.88231	0.706689
2	8	24832149	-1	-1
3	8	24602611	0.826754	0.350603
3	8	24832149	-1	-1
4	8	24602611	0.706323	0.497259
4	8	24832149	-1	-1
5	8	24602611	0.436769	0.434759
5	8	24832149	-1	-1
6	8	24602611	0.296601	0.427906
6	8	24832149	-1	-1
7	8	24602611	-1	-1

Figure 4-2: Data from “GameBall”

The “Player” table contains information regarding the users on the system. A row in the “Player” table is created using the website, but the table itself is utilized on both the Vision System and the website to create a seamless experience. The “passwordEncrypted” column always contain an 84 char long hashed sequence, which ensures the users password cannot be accessed by any outsiders. The “eMail” and “displayName” remains unhashed. The process, and details of, hashing will be further explored in chapter 5.3. Each player starts with 1200 “mmr”, 0 “wins”, and 0 “games”. A display of player data can be seen in Figure 4-3.



playerId	passwordEncrypted	eMail	displayName	mmr	wins	games
2	AQAAAAEAACcQAAAAE...	h@usn.no	Henrik	1411.254956	8	10
12	AQAAAAEAACcQAAAAE...	jjjj@usn.no	Audun	1409.064819	8	10
8	AQAAAAEAACcQAAAAE...	hhhh@usn.no	Emil	1407.281159	8	10
10	AQAAAAEAACcQAAAAE...	jj@usn.no	Simen	1371.698577	7	9
9	AQAAAAEAACcQAAAAE...	j@usn.no	Birk	1361.110519	7	10
14	AQAAAAEAACcQAAAAEF...	ll@usn.no	Adrian	1347.349003	7	11
3	AQAAAAEAACcQAAAAEE...	kk@usn.no	Johan	1332.322587	7	12
11	AQAAAAEAACcQAAAAE...	jjj@usn.no	Leon	1322.396874	7	13
15	AQAAAAEAACcQAAAAEP...	lll@usn.no	Haakon	1304.438684	4	5
4	AQAAAAEAACcQAAAAEJ...	kkk@usn.no	Tobias	1289.614959	4	6
7	AQAAAAEAACcQAAAAE...	hhh@usn.no	Samuel	1273.625608	4	7
1	AQAAAAEAACcQAAAAEP...	k@usn.no	Kevin	1258.68176	4	8
13	AQAAAAEAACcQAAAAEE...	l@usn.no	Petter	1229.73086	2	4
16	AQAAAAEAACcQAAAAEP...	llll@usn.no	Harald	1200		
6	AQAAAAEAACcQAAAAE...	hh@usn.no	Jonathan	1200		
5	AQAAAAEAACcQAAAAE...	kkkk@usn.no	Jonas	1160.009914	0	1

Figure 4-3: Data from “Player”

5 Website

The website is meant to operate as the main GUI. The LabVIEW application will act as the brain of the Vision System, while the website will offer necessary functionalities for the users, as well as displaying data. A user guide for the website can be viewed in Appendix F.

5.1 Use Case

Figure 5-1 below shows a Use Case diagram of the website. The Use Case is an illustration summarizing all functions, as well as role differences, in the website. There are no intended “operator” or “admin” role, therefore, all actions will be available to all users, although some functions require an account on the website.

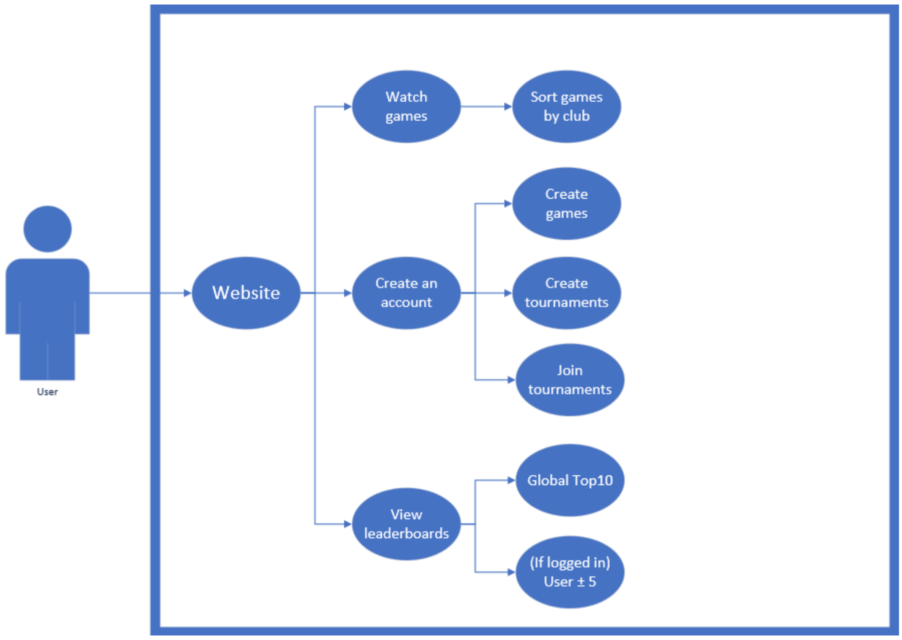


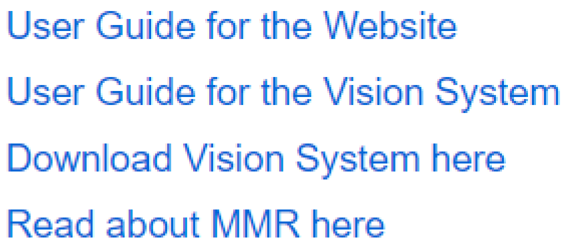
Figure 5-1: Use Case Diagram

As seen in Figure 5-1, the user is able to watch games, create their own account, and view leaderboards. When viewing a game, the games are sorted by which club the game is played in. By creating an account, most of the essential functions are allowed, such as creating games and tournaments, as well as joining tournaments created by other users.

Leaderboards are separated into two categories: Global- and user leaderboards. The global leaderboard is based on the top ten players, while the user leaderboards show the users placement, as well as the five ranks above and below. These rankings are based on the MMR of each player.

5.2 Web pages

The website consists of six web pages, with three of them changing depending on whether the user is logged in or not. The main page, “Home”, is planned to mostly display information rather than having any functional use. In addition to providing information, user guides for both the website and the LabVIEW application and a PDF document about the MMR will be added to the main page, as well an .exe file for the LabVIEW application. These links are displayed in Figure 5-2. Each web page can be accessed through the header shown in Figure 5-3.



User Guide for the Website

User Guide for the Vision System

Download Vision System here

Read about MMR here

Figure 5-2: Download Links

Figure 5-3: Website Header

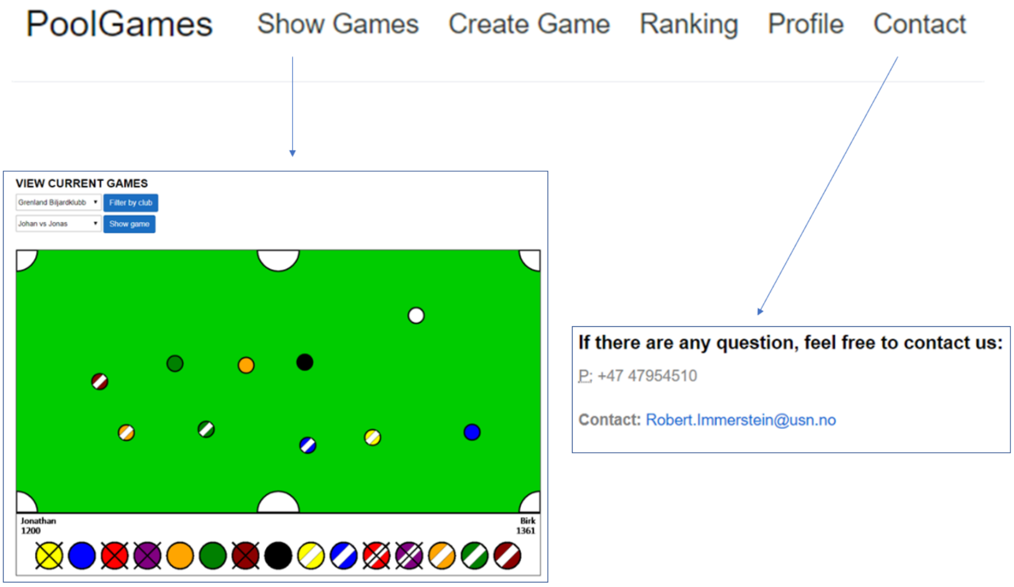
5.2.1 Show Games and Contact

Two of the three web pages which does not differ depending on the login status are the “Show Games” and “Contact” pages. While the “Contact” page primarily offers a contact possibility, the “Show Games” operates as one of the primary attractions of the website.

The “Show Games” page extracts the coordinates for each ball, from the database, and displays each ball accordingly on a canvas. This is done by selecting a game through the two filter inputs above the canvas. The first sort is based on selecting a club from the first drop-down menu. After selecting a club, all active games from the chosen club is shown in the second drop-down menu. A game is active upon creation, and deactivated upon completing the game, in the LabVIEW application. In addition to displaying the pool balls, a display below the pool table, in the canvas, shows which players are playing, the players’ MMR, which balls have been removed, and which balls are still in play.

All drawings are created with JavaScript, while the data is extracted from the database using the ASP.Net Core interface.

Both the “Contact” and the “Show Games” page, as well as the navigation to those pages, are shown in Figure 5-4.



If there are any question, feel free to contact us:

P: +47 47954510

Contact: Robert.Immerstein@usn.no

Figure 5-4: “Show Games” and “Contact” Pages

5.2.2 Profile

In order to access more functionality, an account is needed. For a user to access their account, the user must utilize the “Profile” page. By logging in, the user gains the functionalities in the “Create Game” page, as well as user-based data on the “Ranking” page. Upon loading the “Profile” page, the user has the option to log in, or create an account. To log in, an e-mail and a password is required. If the user does not have an account, the “Create a new user” link, below, can be pressed. This link displays a modal window in which one can create a new user by entering an e-mail, a password, as well as a nickname. When a new account is created, the data is stored in the database, which enables the option to log in. When the user has logged in, a header will display the nickname in a welcome sign. The full display, as well as the navigation, of the “Profile” page is shown in Figure 5-5.

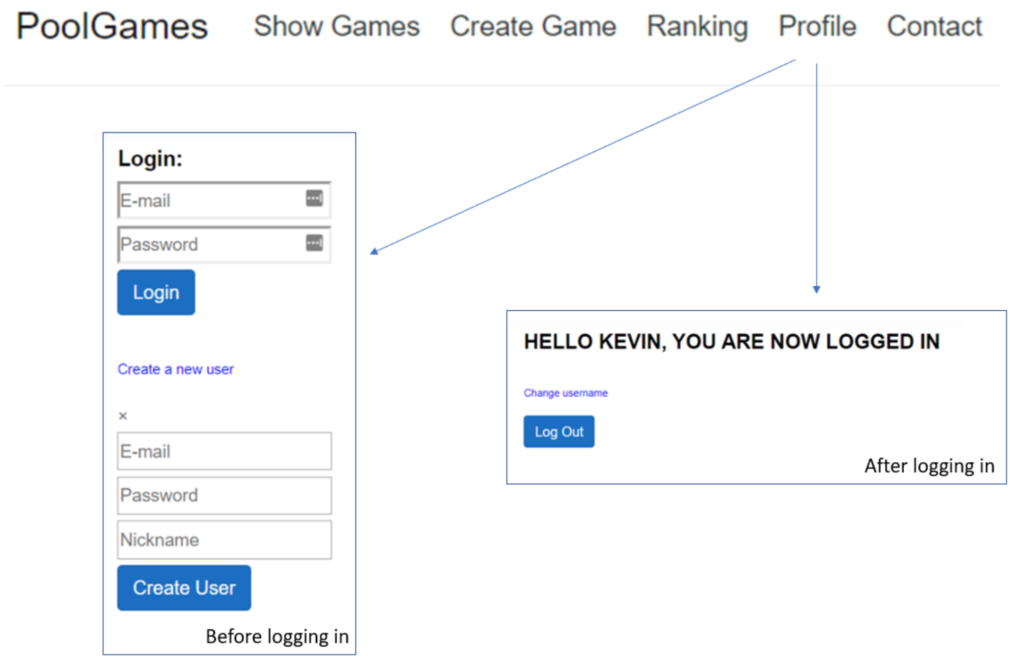
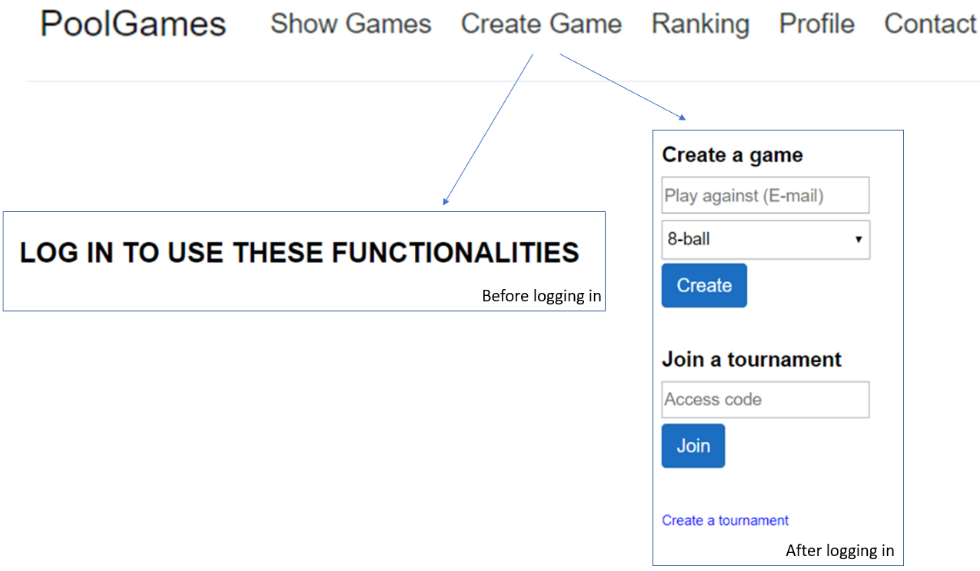


Figure 5-5: “Profile” Page

5.2.3 Create Game

Upon entering the “Create Game” web page, a header will display “Log in to use these functionalities” if the user is not logged in. When the user then logs in, as described in chapter 5.2.2, the page offers the necessary functionality to be able to use the LabVIEW application. To create a game, the user must write the e-mail of the user they wish to challenge in the “Play against (E-mail)” field. After inserting the e-mail, the user then chooses a ruleset from a drop-down menu. After a correct e-mail is written, and a ruleset is chosen, a game can be created by pressing the “Create Game” button. The web page will then display a unique four-character long access code, which can then be entered into the LabVIEW application, to start the game. In addition to creating a game, the “Create Game” page also offers the possibility to create, and join, tournaments. To create a tournament, the “Create tournament” link can be pressed. This opens a modal window in which the user can press a “Create” button. This creates a tournament with the time stamp as a unique identifier, in addition to the automatically generated “tournamentId” in the database. Upon creating a tournament, a unique eight-character long access code is generated and is displayed below the “Create” button. This access code can be typed into the “Access code” input field, underneath the “Join a tournament” text, to join the created tournament. The creator of the tournament is automatically added to the tournament upon creation; therefore, this feature is mainly for others to be able to join the tournament. If a user is already added to the tournament and tries to join, a “You’ve already joined this tournament” disclaimer will appear. A display of the “Create Game” page, both before and after logging in, as well as the navigation to the page, is shown in Figure 5-6.

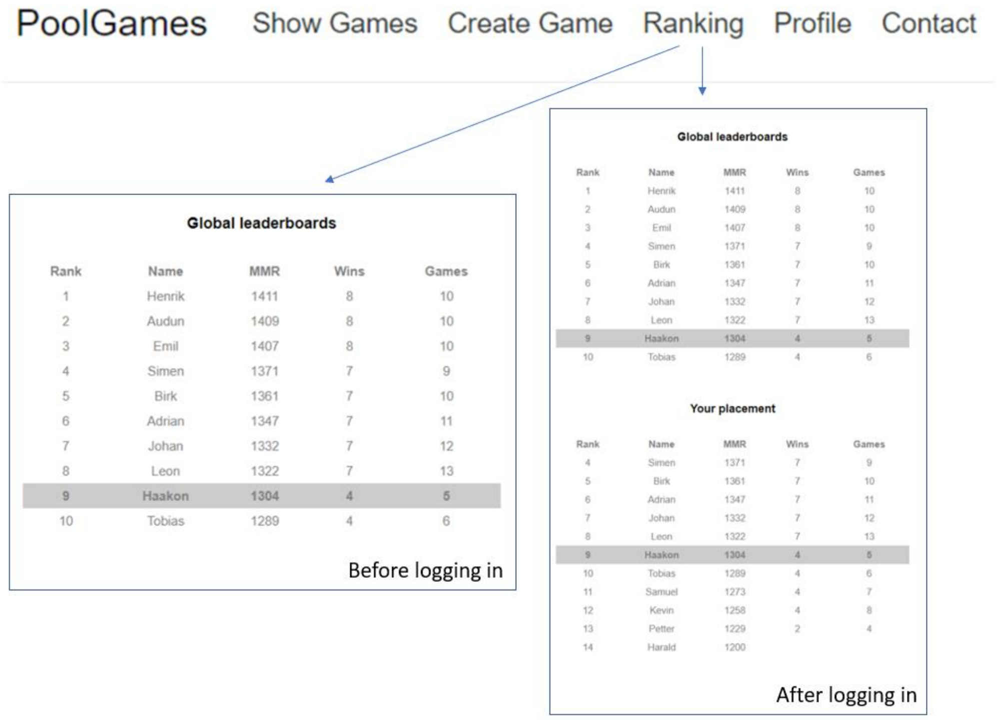


LOG IN TO USE THESE FUNCTIONALITIES

Figure 5-6: “Create Game” Page

5.2.4 Ranking

The “Ranking” page is used to display statistics on the top players, as well as displaying the user’s statistics. In addition to displaying the statistics of the user, the five users with ranking directly higher, and five directly lower, are displayed as well. The “Global leaderboards” are displayed at all times, while the “Your placement” leaderboard is added to the page when the user has logged in. The leaderboards display the rank, nickname, MMR, number of wins, and total number of games, for each player. All data is extracted from the database. A display of the “Ranking” page, both when logged in and not, is shown in Figure 5-7.



Global leaderboards

Rank	Name	MMR	Wins	Games
1	Henrik	1411	8	10
2	Audun	1409	8	10
3	Emil	1407	8	10
4	Simen	1371	7	9
5	Birk	1361	7	10
6	Adrian	1347	7	11
7	Johan	1332	7	12
8	Leon	1322	7	13
9	Haakon	1304	4	5
10	Tobias	1289	4	6

Before logging in

Global leaderboards

Rank	Name	MMR	Wins	Games
1	Henrik	1411	8	10
2	Audun	1409	8	10
3	Emil	1407	8	10
4	Simen	1371	7	9
5	Birk	1361	7	10
6	Adrian	1347	7	11
7	Johan	1332	7	12
8	Leon	1322	7	13
9	Haakon	1304	4	5
10	Tobias	1289	4	6

Your placement

Rank	Name	MMR	Wins	Games
4	Simen	1371	7	9
5	Birk	1361	7	10
6	Adrian	1347	7	11
7	Johan	1332	7	12
8	Leon	1322	7	13
9	Haakon	1304	4	5
10	Tobias	1289	4	6
11	Samuel	1273	4	7
12	Kevin	1258	4	8
13	Petter	1229	2	4
14	Harald	1200		

After logging in

Figure 5-7: “Ranking” Page

5.3 Password Hashing

Password hashing is done by encrypting the password based on combining the password and e-mail. In addition to the encryption, the password is combined with a static sequence, also called a salt, to further encrypt the password. This is done immediately upon creating an account, and the encrypted password is stored in the database. When attempting to log in, the typed in password is encrypted, in the same fashion as mentioned above, and the encrypted password is compared to the encrypted password retrieved from the database. If the passwords match, the login is accepted.

6 Discussion

This discussion chapter contains a short evaluation on the methods and solutions the group has provided in different parts of this report. The first subchapter contains discussions on the Vision System and some quirks on its implementation, the second subchapter speaks of the website, and the final subchapter discusses future development.

6.1 Vision System

The goal of the Vision System was to create a system which identifies pool balls on the pool table, judges the game, and sends the data to a cloud-based database. The system performs as expected, but for optimal performance the system carries with it a few assumptions:

- A minimum of 3 cm distance between the pool balls on the table
- Striped balls must have their stripes angled towards the camera.

The balls require a 3 cm gap because of the particle analysis method in LabVIEW cannot discern between balls if they are too close. If two, or more, balls are closer than 3 cm the Vision System might not register them as individual balls, but rather as one large ball.

Because the pool cloth is blue, identifying the blue balls posed a challenge. Classifying a pool ball requires an accurate, and precise, ROI. Figure 6-1 shows the blue balls having a very imprecise ROI, which makes analyzing the color spectrum within the ROI different from the other pool balls' ROI. The striped blue ball suffers greatly from this due to the percentage of white color in the color spectrum being much lower than usual.

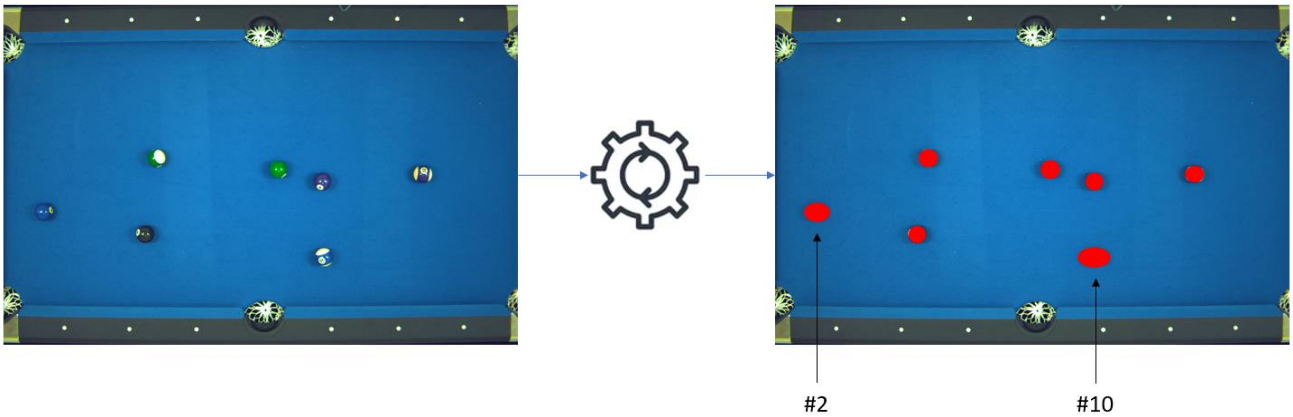


Figure 6-1: Identification of Balls #2 and #10

The angle of the striped balls is important for accurate classification, because the color spectrum analysis require the white part of the stripe to identify the pool ball as striped. This introduces an assumption in the Vision System requiring striped balls to show at least part of their stripe to the camera. This is illustrated in Figure 6-2 where two different circumstances, with two different sets of orientations, on the same balls affects the Vision System's ability to properly identify the pool balls.

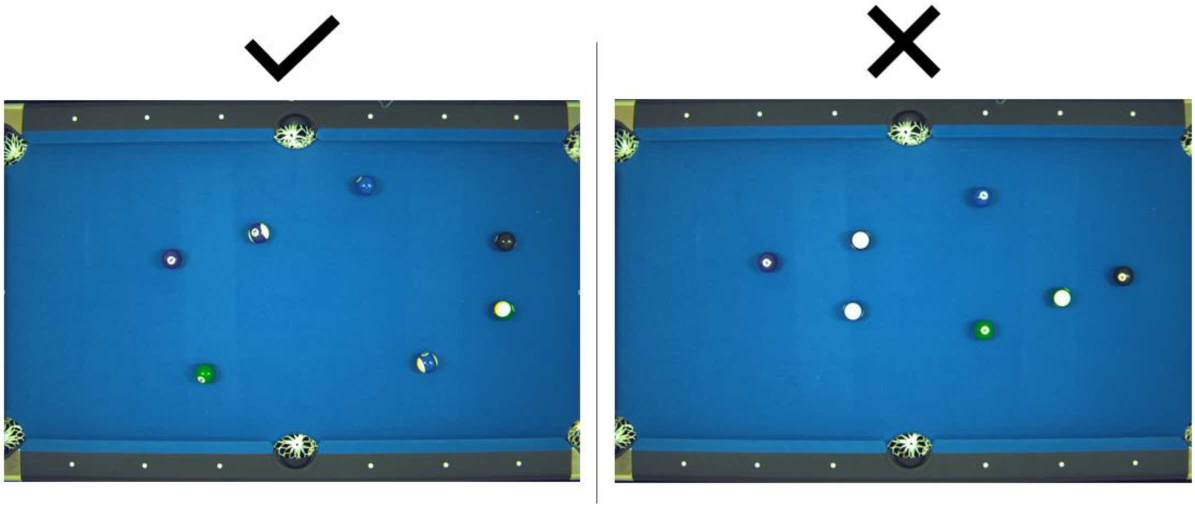


Figure 6-2: Striped Angle Consequences

6.2 Website

The initial goal of the website was to display data such as statistics, current games, historical games, and general information about pool. A Use Case diagram was created at the start of the project, where many of the goals quickly turned out to be improbable when faced with the strict necessities of the Vision System, as well as the group's preferences changing.

In terms of watching games, the initial ambitious goal was to watch live-streamed footage, as well as storing the live-streams to watch later. Due to a lack of streaming possibilities, as well as switching to a picture-based application, it was easier to draw the table onto a canvas based on the coordinates of the balls in play. It was decided early on to not focus on storing previous games as well, due to a lack of interest in storing cartoon adaptations; however, the game data for historical games are still stored, which allows for historical games in a future version. In addition, the name and MMR of the players of the game are displayed, but the placement of the player names does not reflect their team, i.e. the system does not save which player plays which type of ball, only which player wins.

Originally, the possibility of creating an account was planned to only be necessary to view personal statistics, game results, and creating personalized rulesets for games. Due to limiting factors in the Vision System, neither game results, other than documenting the winner, nor the implementation of personalized rulesets, due to a lack an advanced judge, has been implemented in the account functionality. An option to choose between rulesets, stored in the database, has been added, but this choice does not affect the game in any way.

The option to create, and join, tournaments have been added, but the tournament element has not been fleshed out. Creating, and joining, a tournament offers no functionality other than adding data to the database. Although the tournament functionality was never built upon, the feature was kept in both the database, and the website, as a future possibility, and as a reminder of a development option if there would be time to add upon the goals for the system.

The leaderboards have been implemented just as planned, although the user-based leaderboard has been expanded to displaying the five players above, and the five players below, the user, rather than the documented "+-2" shown in Figure 6-3.

Even in the original Use Case diagram, shown in Figure 6-3, the possibility of a forum was documented with doubts of realization. This was not considered a priority, and while it is not considered a difficult feature to add, the amount of time needed to code the forum, add tables to the database, as well as testing the forum made the forum a low priority seeing as time was the constraining factor.

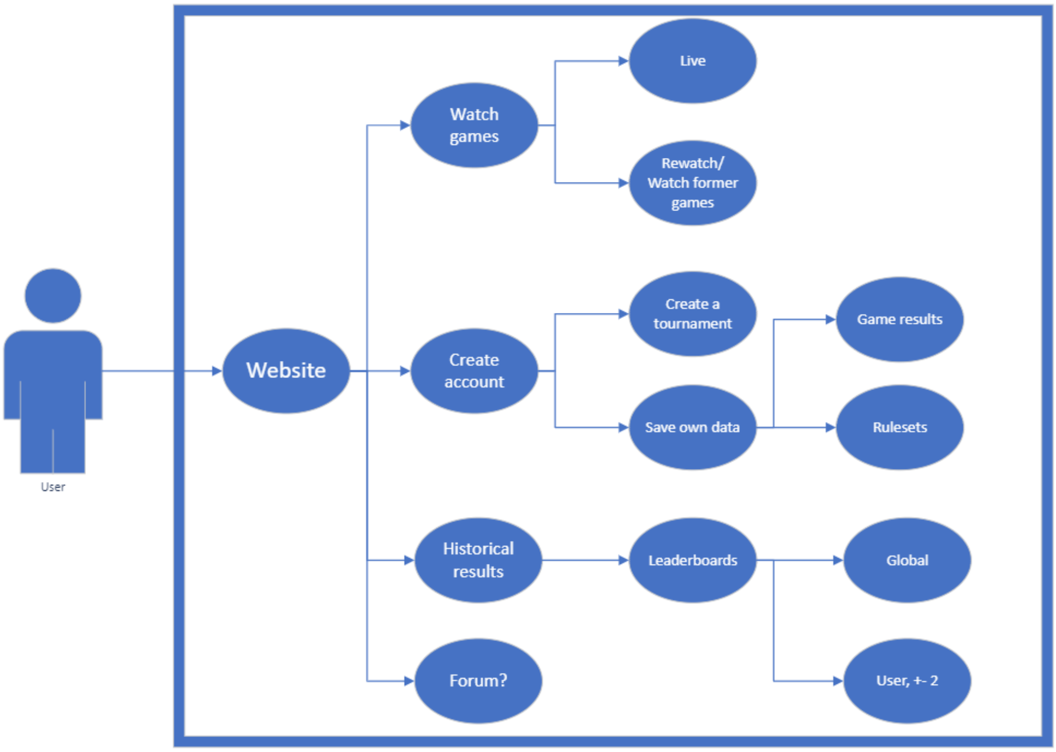


Figure 6-3: Original Use Case Diagram

6.3 Future Development

This project is meant as a prototype towards a finished product. This subchapter will discuss the future development of the prototype, and the problems present in the prototype.

The first set of problems are related to the Vision System's assumption of the game. Two, or more, pool balls are often found sitting right next to each other, and striped balls can sometimes be found with their stripes horizontal.

Another problem to solve, related to the Vision System, is the color of the cloth on the pool table. The Vision System in the solution is a static system, meaning it will not change based on its environment. The cloth the system is designed for is blue, however there are multiple different colors found all over the world, which may pose a problem.

The Vision System’s most limiting factor is the fact the it only uses one camera. This one camera might not show everything correctly. One possible solution to this is using multiple cameras with different positions and angles. Using multiple cameras to find the pool balls might provide better precision and remove the assumption on the striped balls. Another argument for multiple cameras is the streaming aspect of the application. Currently, the balls are shown on a canvas on the website; but preferably, a livestream can be set up using multiple angles to show the game.

The website can show currently active games being played, but as soon as the game is concluded, the game is unwatchable. This is not because it is deleted, but because the feature was never added. The system is set up to allow it, but time constraint limited the feature set in this first prototype.

This entire project has been solved using the groups ideas on how to solve the problems presented. Machine learning might solve a lot of the problems presented in a better way than any human could, but machine learning was never the focus of this project.

The database stores some identifying information on the players registering on the website. This data is never deleted, something which should be allowed in later versions. A General Data Protection Regulation (GDPR) notice and documentation should be available in later versions. It is not currently possible to change neither the e-mail, nor the password, after creating an account. Deleting an account, using the website, is not possible either.

The website should display active access codes and what those codes refer to in a separate page for users. This allows users to leave the site without losing their access code permanently.

7 Summary

The goal of the project was to create a system capable of presenting the pool games played. To achieve this goal, a Vision System was created that takes in image data of a pool game and returns the game state data. The Vision System is the judge for the games being played, accounting for balls being shot down, and which player plays which type of balls. The games played are presented locally through the Vision System and online with a website created during the project. The website functions as a mediator between the user and the system. The website reads data from the Vision System, displays data, takes in user input, and makes it possible to interact with the system.

The Vision System is created in LabVIEW because of the existing vision tools already present in the LabVIEW ecosystem. Because of the existing ecosystem in LabVIEW, C# as a development possibility was discarded. Several cameras were tested, and it was concluded that the Basler camera was the best fit for this project. During development, the team only had access to one scenario regarding the light of the surroundings and color of the cloth on the pool table. These limiting factors made ensuring quality service outside of the development parameters uncertain. Because of the uncertainty regarding the Vision System in unknown circumstances, widespread use of this system might not be optimal.

The website was created using ASP.Net Core 3.1 because of the experience in the programming languages utilized prior to starting up development. The website was intended as a place to view ongoing, and concluded, games, and tracking personal progress. As development went on, the option of moving functionality from the Vision System to the website presented itself and was preferable to simplify the Vision System's user interaction and limiting the possibility of a data breach regarding user passwords.

The integration level between the Vision System, and the website, has evolved throughout this project, and has now reached a point where one part cannot be used without the other. Initially, each part could be utilized individually, but has since reached a symbiotic relationship where the website manages the user inputs, while the Vision System handles the games being played.

Although the group had a lot of ambition when starting work on this project, it was quickly toned down to accommodate the most limiting factor during this project, time. The goal quickly went from developing a feature complete solution, into developing a prototype allowing further development down the road.

8 References

[1] “Intel,” Intel Corporation, [Online]. Available: https://www.intel.com/content/www/us/en/products/boards-kits/nuc.html. [Accessed 1 May 2020].

[2] K.-S. Kwon and S. Ready, Practical Guide to Machine Vision Software : An Introduction with LabVIEW, Weinheim: Wiley-VHC, 2015.

Appendices

Appendix A – Project Description

Appendix B – Basler acA5472-17uc Settings

Appendix C – Application GUI & User Guide

Appendix D – Image Processing Sequences

Appendix E – Match Making Ranking Equations

Appendix F – Website User Guide

Appendix A

Project Description

University of
South-Eastern Norway

FM4017 Project

Title: **Implementation of Cloud-based Vision System with focus on Internet of Things, Web Technology, Machine Learning and Cyber Security**

USN supervisor: Hans-Petter Halvorsen, Maths Halstensen and Robert Immerstein

External partner: Grenland Biljardklubb

Task background:

This research and development project will combine and use many of the central topics learned in the IIA Master study program. You will combine topics like Internet of Things, Database Systems, Software Engineering, Web Technology, Sensor Technology and Machine Learning and issues regarding Cyber Security. You will get a wider theoretical view of many of these topics combined with practical implementation and realization.

See Figure 1 for a suggested system overview. You are relatively free to choose which parts you want to focus on, what type of technology you want to use, etc.

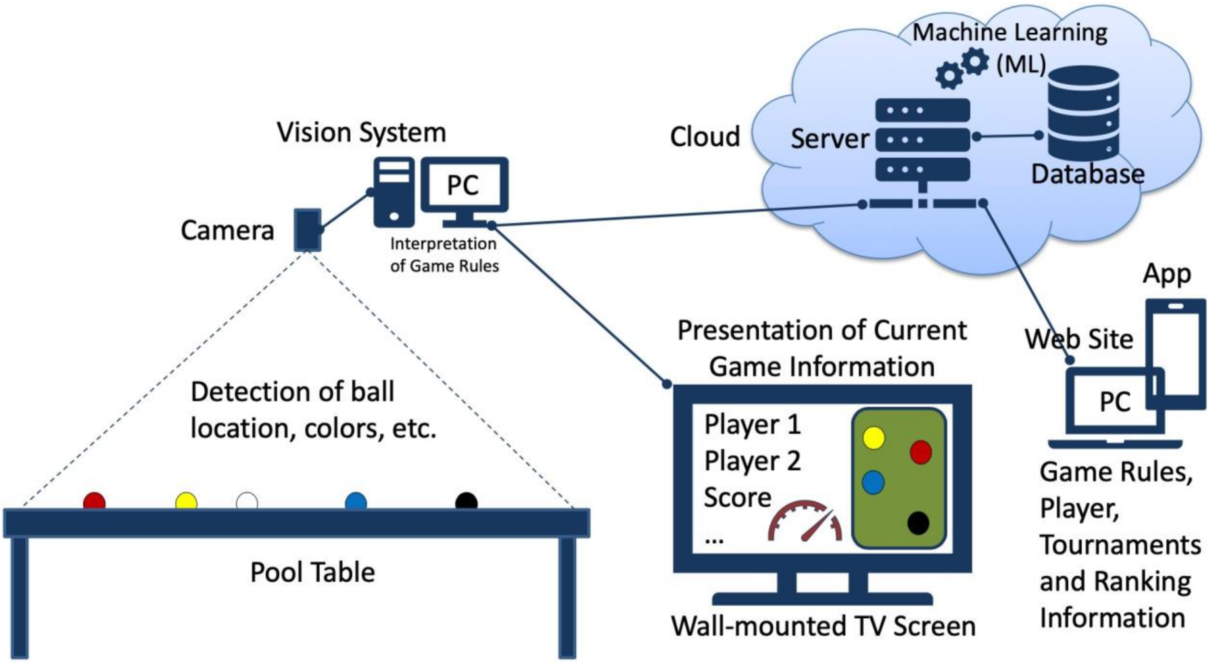

Machine Learning (ML)
Cloud
Server
Database
Vision System
PC
Camera
Interpretation of Game Rules
App
Web Site
PC
Presentation of Current Game Information
Detection of ball location, colors, etc.
Player 1
Player 2
Score
...
Pool Table
Wall-mounted TV Screen
Game Rules, Player, Tournaments and Ranking Information

Figure 1: System Overview

Task description:

In this research and development project we will implement a prototype of a **Cloud-based** pool table computer vision competition system with focus on **IoT**, **Web** Technology, **Machine Learning (ML)**, **Cyber Security**, **Smart Sensors** and **Vision** Technology.

Suggested Project Activities

The following activities are relevant in this project:

- Give an overview of vision systems in general and implement a functional prototype that can be used to detect the positions of the balls, the colours of the balls, etc. An important part of the research and development project is to test out different hardware and software and compare the results achieved
- Vision software: Get an overview of different approach and possibilities when it comes to vision software and programming languages, such as NI Vision Development Software for LabVIEW and C#. Other alternatives may be use of Python.
- Explore different hardware used for the vision system, both different types of cameras, and different types of hardware and operating systems where the software is running, like ordinary PCs, Raspberry Pi, NUC computers, etc.
- Development of vision application prototype.
- Development of application(s) prototype(s) for management and presentation of pool games, results, player information, tournaments Information, ranking Information, etc.
- A Cloud-based Database system should also be design and implemented, e.g. using the Microsoft Azure platform.

- Get an overview of Data Security, GDPR and Cyber Security Threats for such systems. Implement some use cases regarding this.
- See how Machine Learning (ML) can extend the system and implement some use cases regarding this.
- Installation and deployment of prototype at USN and Grenland Biljardklubb

When the project starts, we can together select a proper number of tasks that should be part of this project.

Student category:

IIA or others that are interested in the relevant topics

Practical arrangements:

All necessary facilities and equipment will be available at USN. A separate room at USN with all necessary equipment (including a Pool table, different Vision hardware and software, etc.) will be provided.

Signatures:

Supervisor (date and signature):

Students (date and signature):

Appendix B

Basler acA5472-17uc Settings

The Basler camera has many advanced settings that can be adjusted. Table 0-1 shows the settings that are important to optimize the performance of the Vision System created in this project.

Table 0-1: Basler acA5472-17uc Settings

Attribute	**Value**
Sequence Mode	Off
Width	5472
Height	3648
Offset X	12
Offset Y	12
Pixel Format	Bayer RG 8
Gain Auto	Off
Gain	2,999
Black Level	3
Gamma	1,5
Light Source Preset	Tungsten (2800 Kelvin)
Balance Ratio Selection	Red
Balance Ratio	1,141
Exposure Time	99996
Auto Target Brightness	0,502

Appendix C

Application GUI & User Guide

The LabVIEW application is divided into four steps:

1. Start Game
2. Game Initialization
3. In Game
4. Finish

When the application is executed, the “Start Game” page will display as shown in Figure 0-1. An access code, generated from the website, needs to be entered in the “Access Code” field. After the access code has been written in text field, the game starts by pressing the “Start Game” button. If the entered access code is incorrect, an error will appear as shown in Figure 0-2.

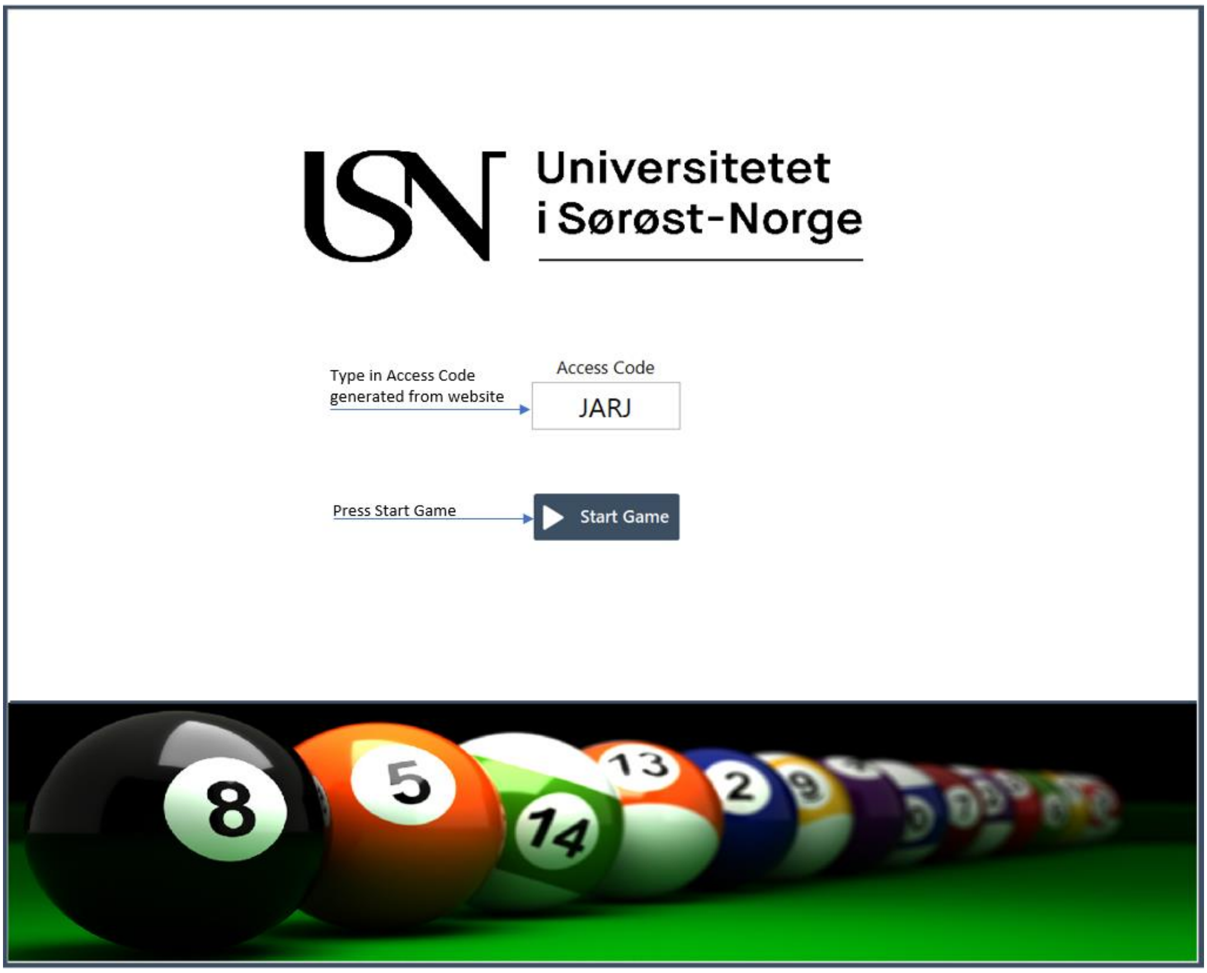
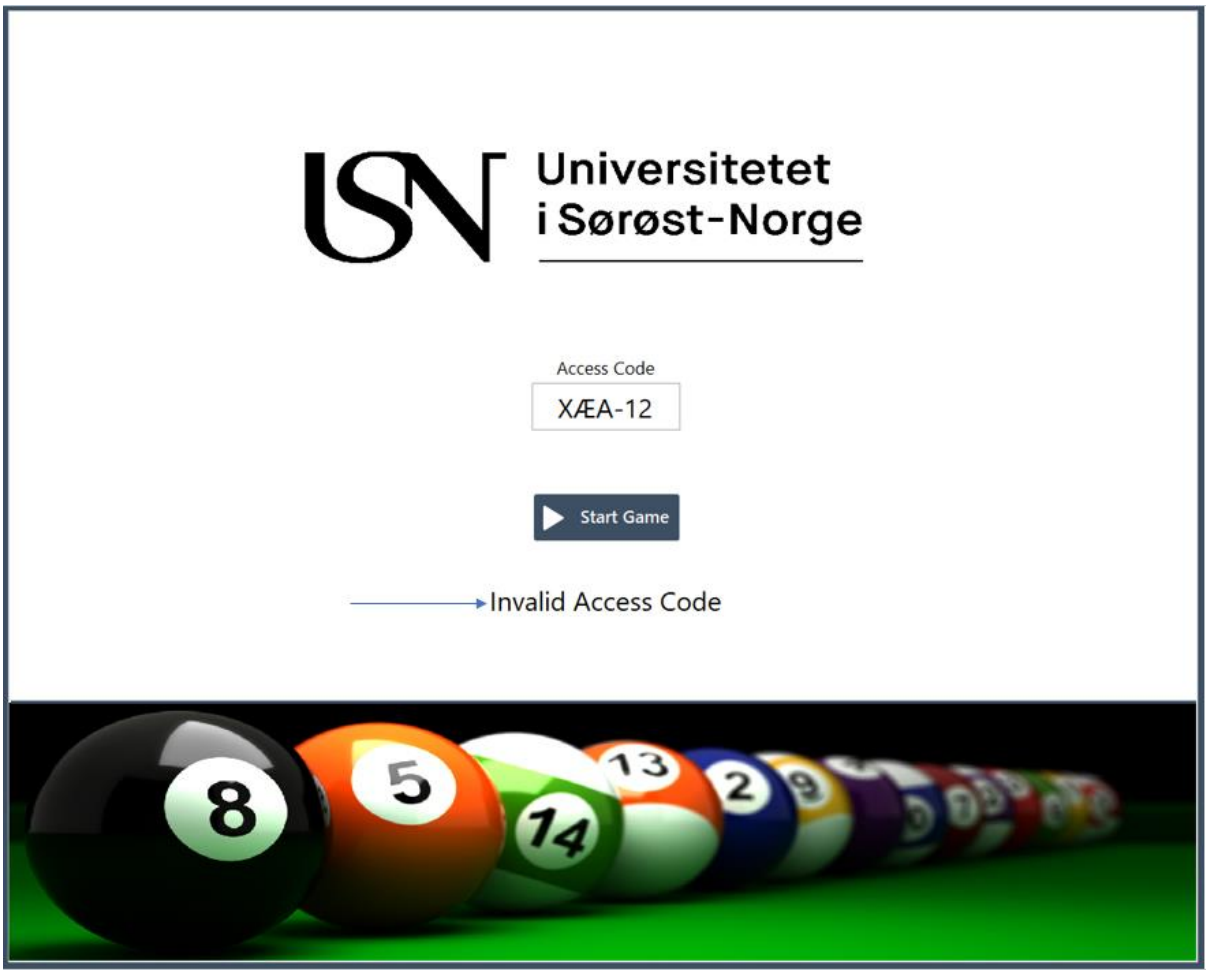


Figure 0-1: Start Game Screen

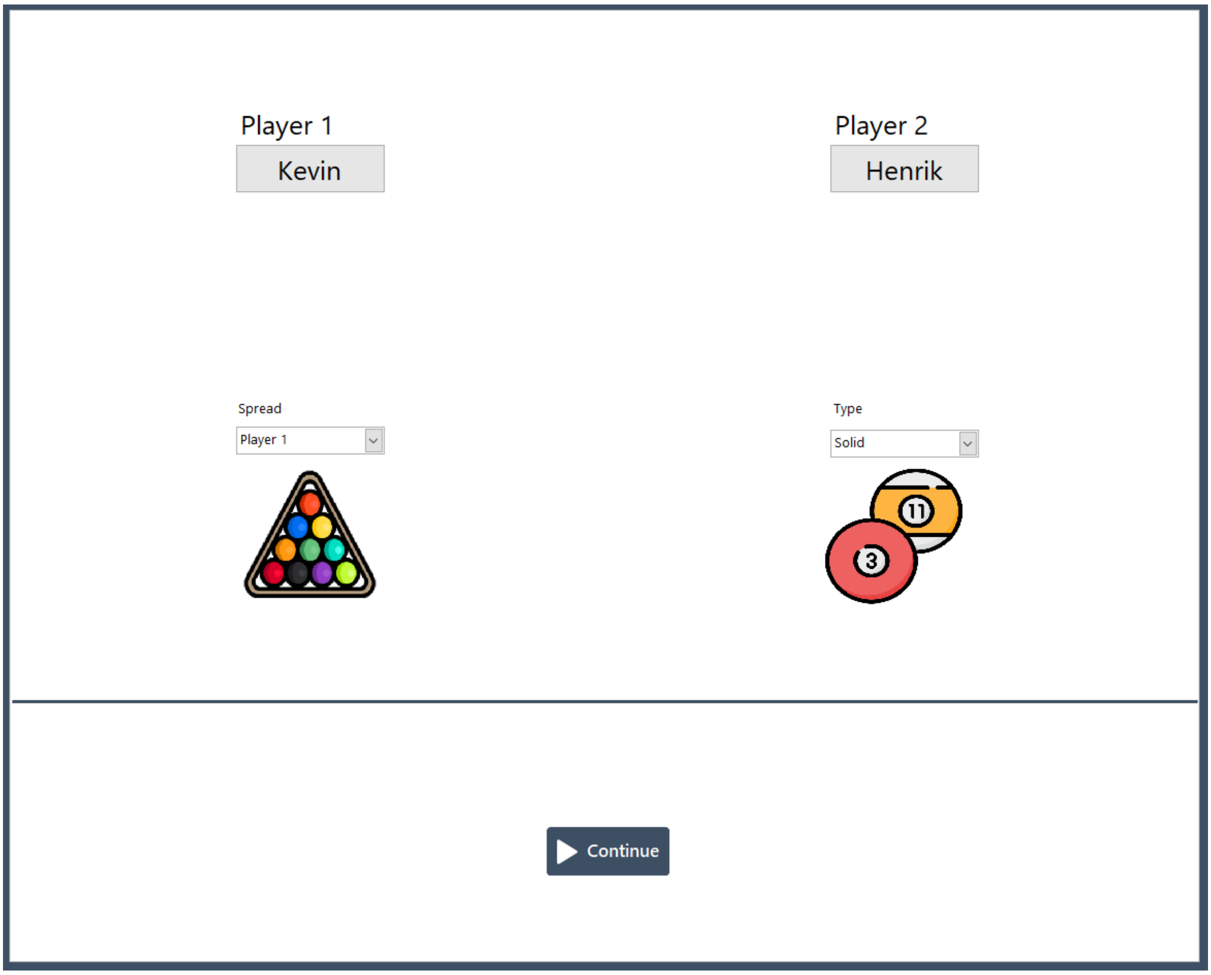


Access Code
XÆA-12

Invalid Access Code

Figure 0-2: Invalid Access Code

When the access code is valid and the “Start Game” button is pressed, the “Game Initialization” page will be shown as in Figure 0-3. This page will display both players’ display name and two dropdown menus, one for spreading and one for pool ball type. Since the application is based on image processing, the application requires that the players choose which player has spread, and which pool ball type the player shot down or chose.



Player 1

Kevin

Player 2

Henrik

Spread: Player 1

Type: Solid

Figure 0-3: Game Initialization Screen

When the game is initialized and the “Continue” button is pressed, the game window will be shown as seen in Figure 0-4. The “Process” button is used to take a snapshot from the ongoing game, and starts the image processing of the snapshot. When running the prototype, a set of images, from an already played game, is already stored in the application folder. The “Process” button will then process one image from the set of images in a predetermined sequence. During a game, the cue will point at the player that is currently shooting. The pool balls that have been shot down during the game will appear below the players’ nickname.

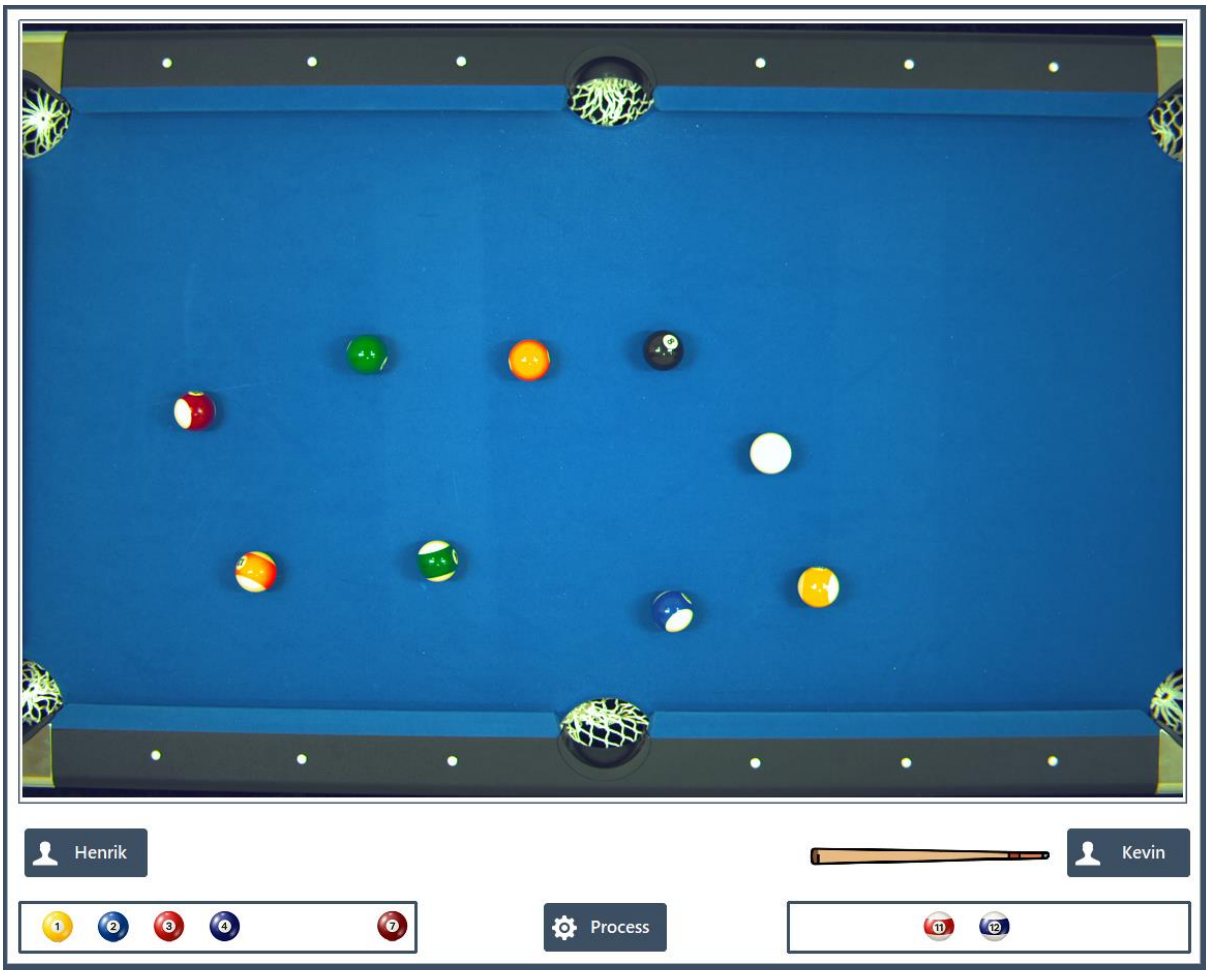


Figure 0-4: In Game Screen

When the black pool ball is downed, the “Finish” window is displayed as seen in Figure 0-5. The winner of the game is displayed on the golden trophy. There are two buttons, one to play again, and one to exit the application. By pressing “Play Again” button, the “Start Game” window will appear, and a new access code, for a new game, is required. The “Exit” button will quit the application.

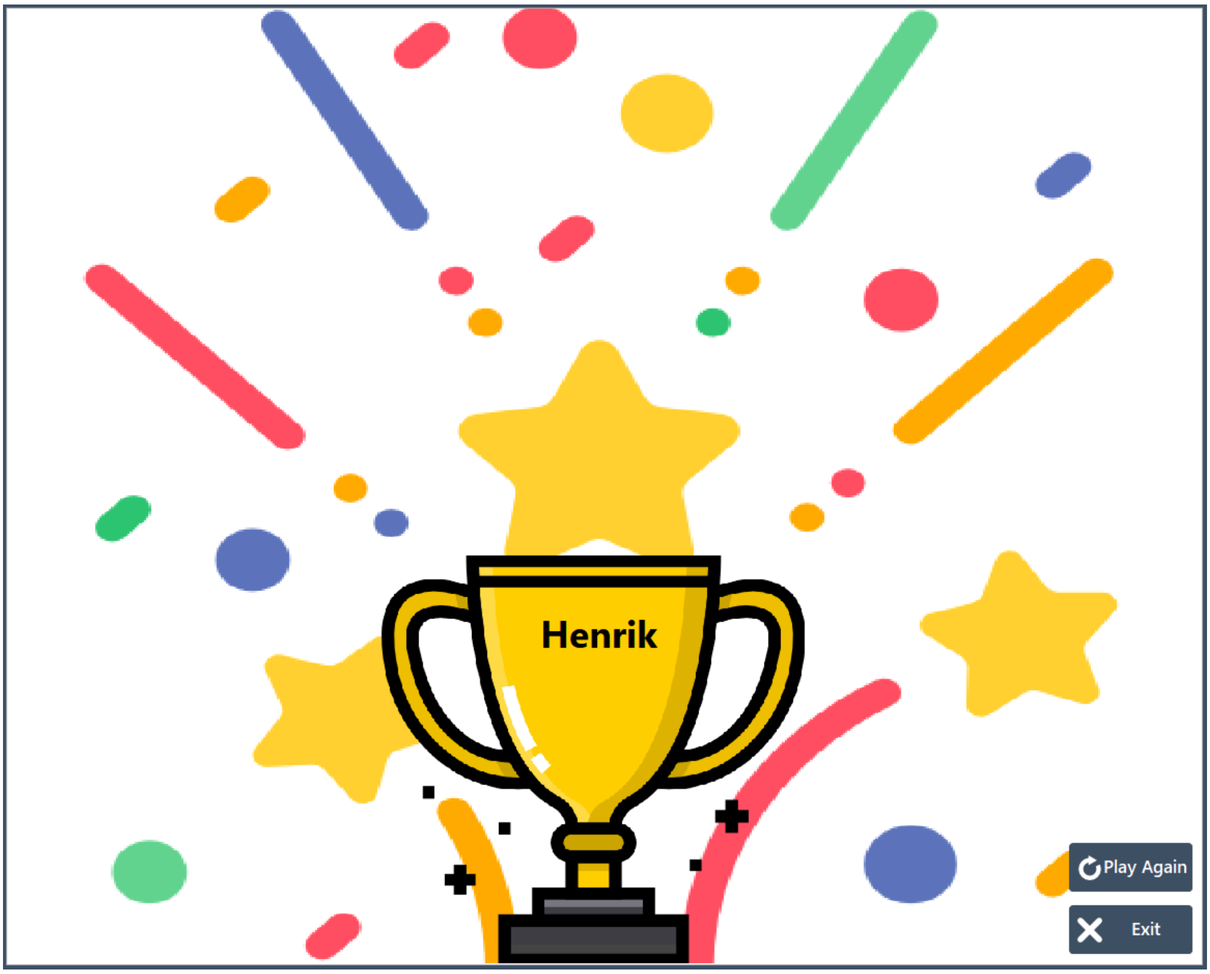


Figure 0-5: Finish Screen

Appendix D

Image Processing Sequences

Four Vision Assistants are needed to extract every pool ball on the pool table. In this appendix, every image processing sequence for the Vision Assistants are illustrated with flowcharts. A list of which ball each Vision Assistant is responsible for extracting is described as well.

Image Processing Sequence 1

This sequence is responsible for extracting the following pool balls:

- White
- Yellow (1)
- Red (3)
- Purple (4)
- Orange (5)
- Wine Red (7)
- Stiped Yellow (9)
- Striped Red (11)
- Striped Purple (12)
- Striped Orange (13)
- Striped Wine Red (15)

The sequence is illustrated with a flowchart in Figure 0-6.

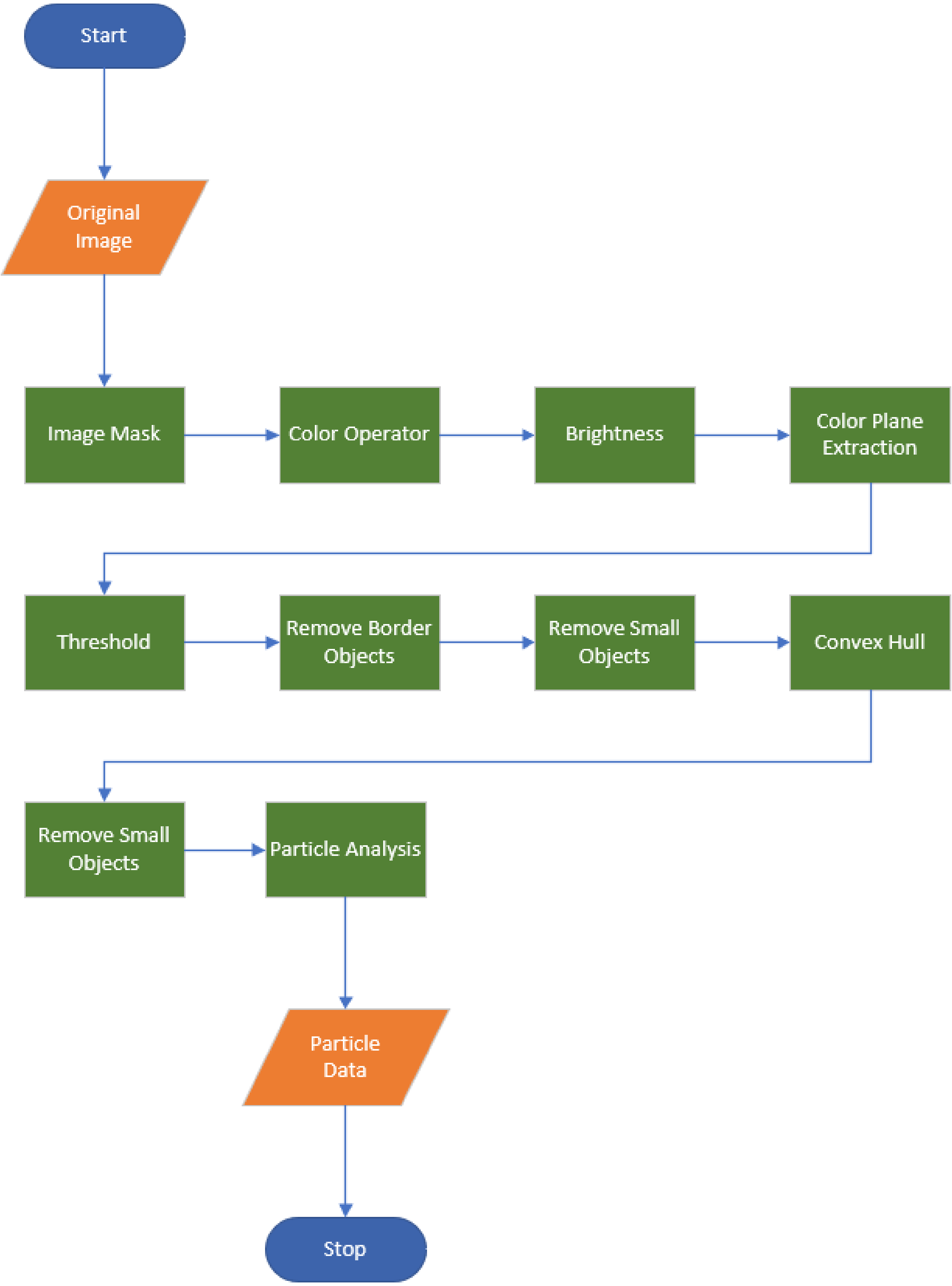


Figure 0-6: Image Processing Sequence 1

Image Processing Sequence 2

This sequence is responsible for extracting the following pool balls:

- Green (6)
- Striped Green (14)

The sequence is illustrated with a flowchart in Figure 0-7.

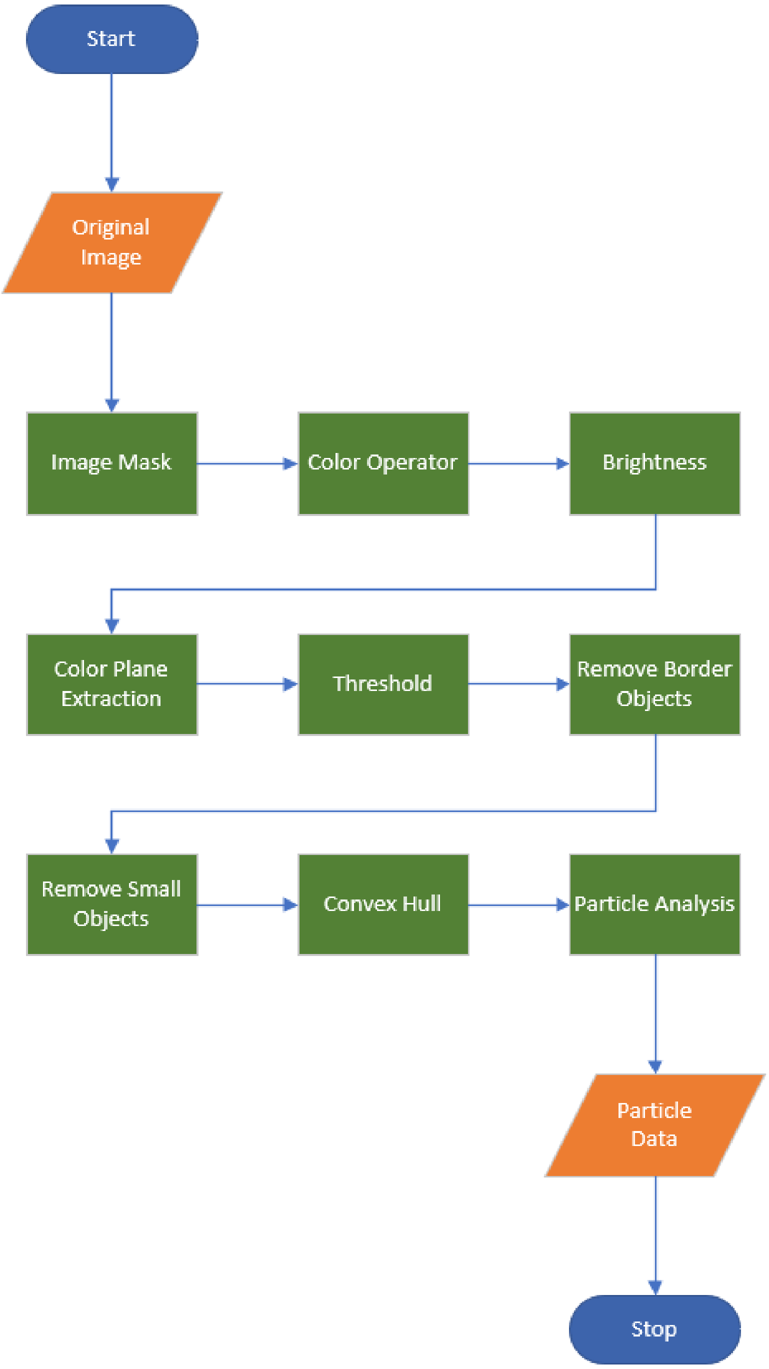


Figure 0-7: Image Processing Sequence 2

Image Processing Sequence 3

This sequence is responsible for extracting the following pool balls:

- Black (8)

The sequence is illustrated with a flowchart in Figure 0-8.

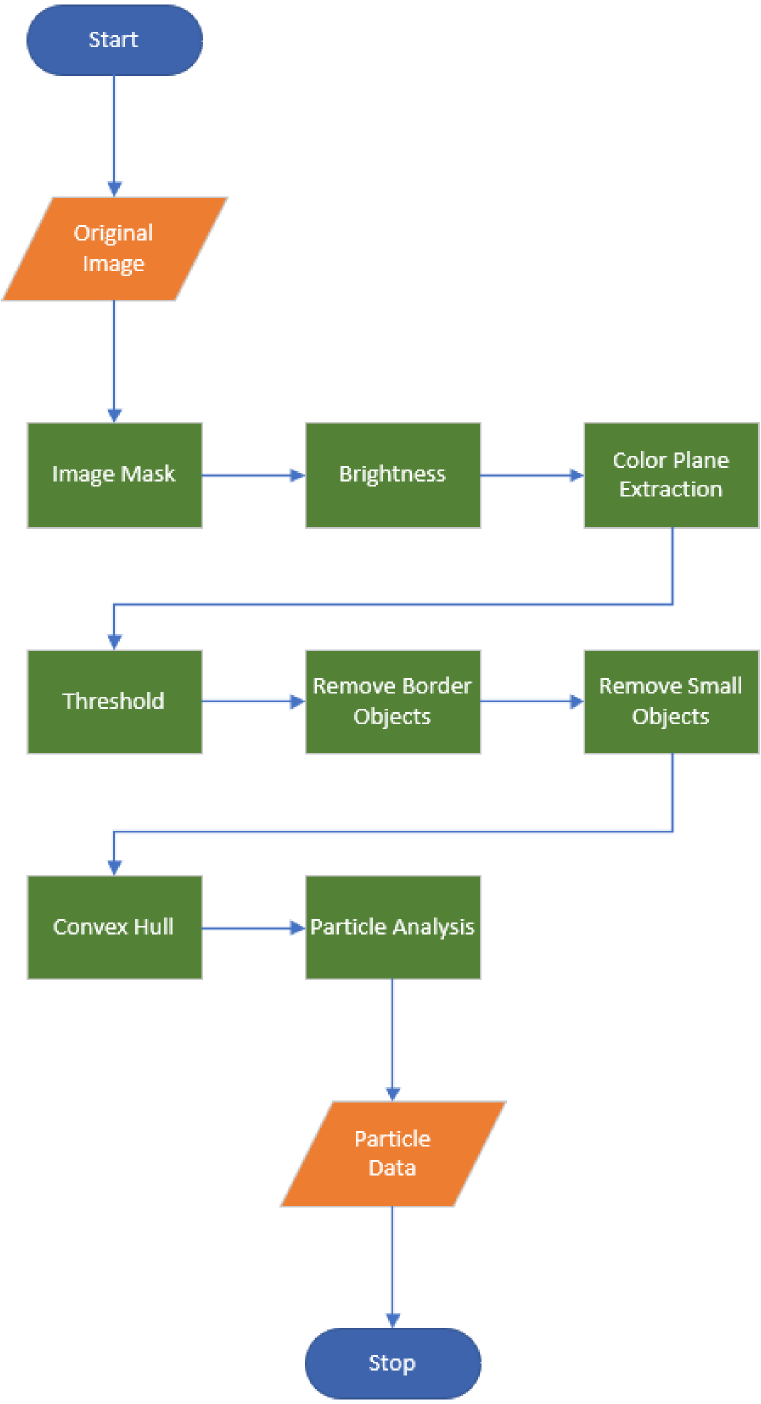


Figure 0-8: Image Processing Sequence 3

Image Processing Sequence 4

This sequence is responsible for extracting the following pool balls:

- Blue (2)
- Striped Blue (10)

The sequence is illustrated with a flowchart in Figure 0-9.

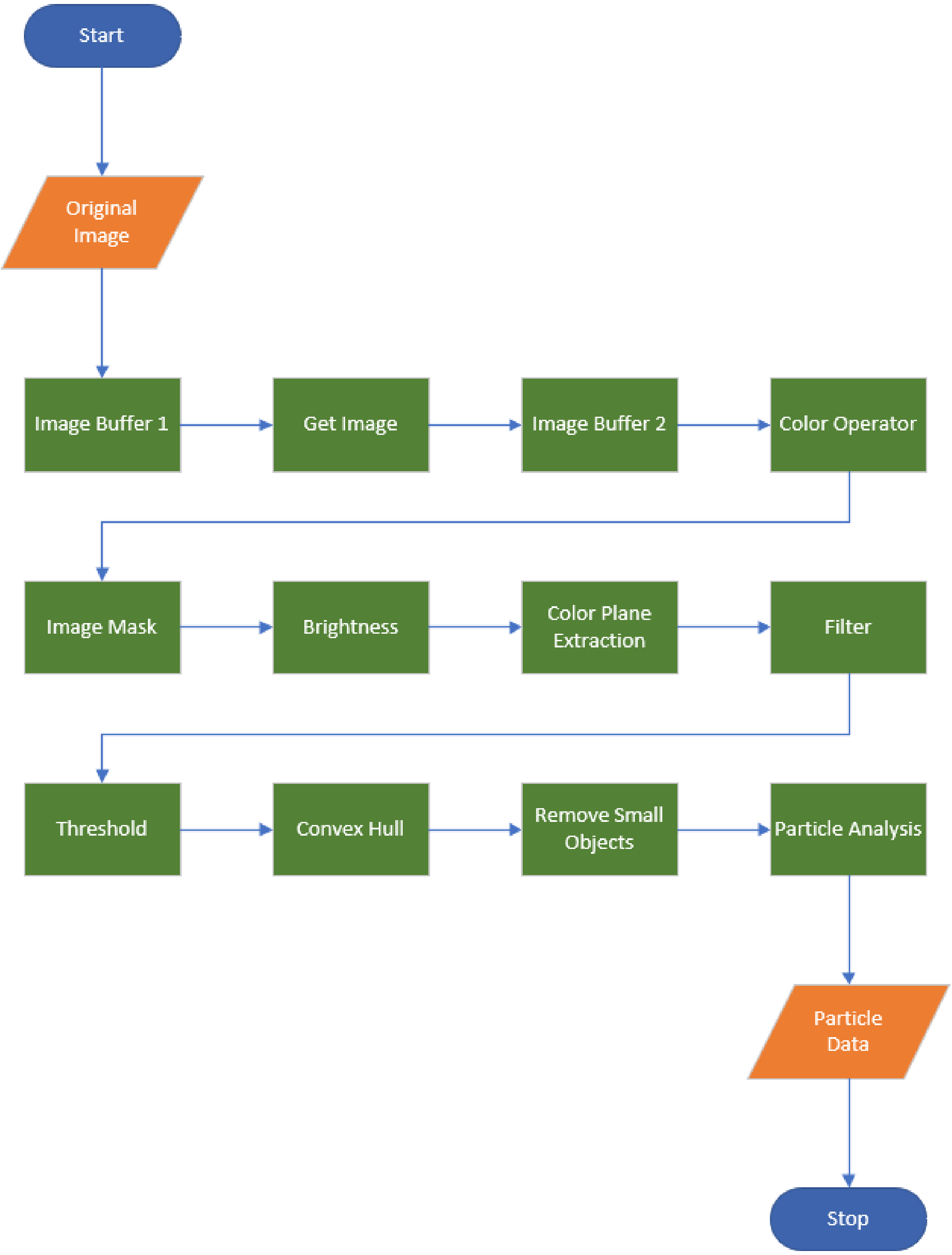


Figure 0-9: Image Processing Sequence 4

Appendix E

Match Making Ranking Equations

To rank the players, a Match Making Ranking (MMR) system was created. The equation used is implemented to create a system capable of taking win rates, disparity, and skill ceilings into account. In chapter 3.5, the MMR system is presented. Equation 1 is the equation that is used to update the MMR of both the winner and the loser when completing a game.

Equation 1: Match Making Ranking Equations

$$ww = total\ number\ of\ wins\ for\ winner$$
$$gw = total\ number\ of\ games\ for\ winner$$
$$wl = total\ number\ of\ wins\ for\ loser$$
$$gl = total\ number\ of\ games\ for\ loser$$

$$\Delta = WinnerMMR - LoserMMR$$

$$UpdatedWinnerMMR = WinnerMMR + \frac{1}{1 + e^{\left(\frac{(-\Delta+439.5)}{200}\right)}} * 400 * \frac{ww}{gw}$$

$$UpdatedLoserMMR = LoserMMR - \frac{1}{1 + e^{\left(\frac{(\Delta+439.5)}{200}\right)}} * 400 * \frac{(gl - wl)}{gl}$$

Appendix F

Website User Guide

A navigation bar is displayed, at the top of the website, and is necessary to be able to navigate between the different web pages. The navigation bar for the website is displayed in Figure 0-10. When in mobile view, the navigation bar is shown as in Figure 0-11. In the mobile view, the different web pages are shown upon pressing the three lines in the upper right corner, also called the “burger”.

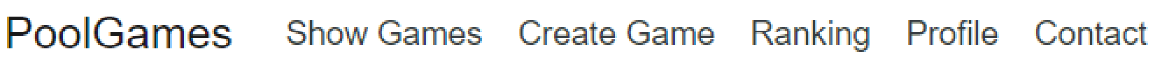


Figure 0-10: A display of the navigation header on the website

Figure 0-11: The navigation bar when using mobile view

Home Page

On the “Home” page, the .exe file for the LabVIEW application, a User Guide for both the website and the LabVIEW application, and a description of the Match Making Ranking (MMR) are displayed. These links are at the bottom of the “Home” page, and the links are displayed in Figure 0-12.

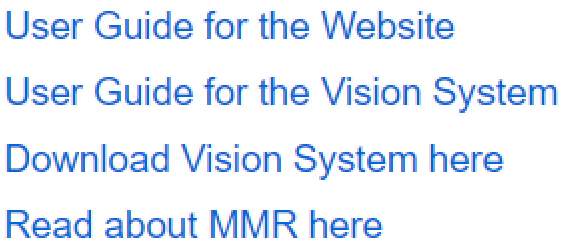
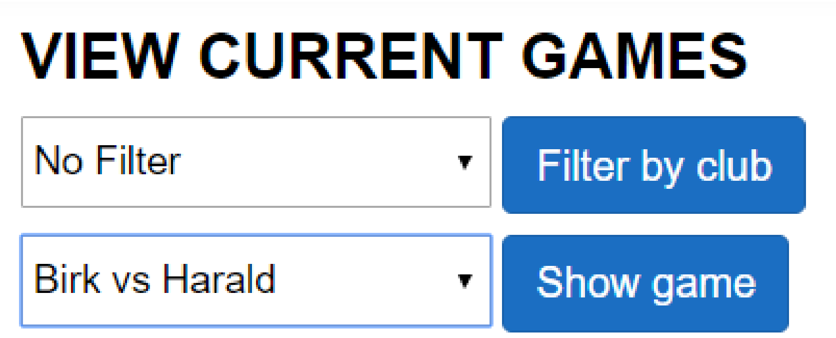

User Guide for the Website
User Guide for the Vision System
Download Vision System here
Read about MMR here

Figure 0-12: The links for user guides, application download, and information on the MMR system

Show Games

On the “Show Games” page, the user can view different games that is currently in play. To view a game, select a club from the upper drop-down menu, shown in Figure 0-13. Then press the “Filter by club” button to the right of the drop-down menu. There is also an option to choose “No Filter” in the drop-down menu. Afterwards, choose a game from the lower drop-down menu. These games are now filtered based on your choice in the “Filter by club”. If “No Filter” was chosen in the previous drop-down menu, all games will appear in this drop-down menu. Choose a game and press the “Show game” button. The current status of the game will then be displayed onto the green canvas screen below the filters.



VIEW CURRENT GAMES

No Filter ▾

Birk vs Harald ▾

Figure 0-13: The filter options

Below the green table on the canvas, the competitor names and Match Making Rankings are displayed. Below the Match Making Ranking, an array of pool balls is displayed with a potential cross over them to indicate them missing from the pool table. The pool table, and contents below, is showed in Figure 0-14.

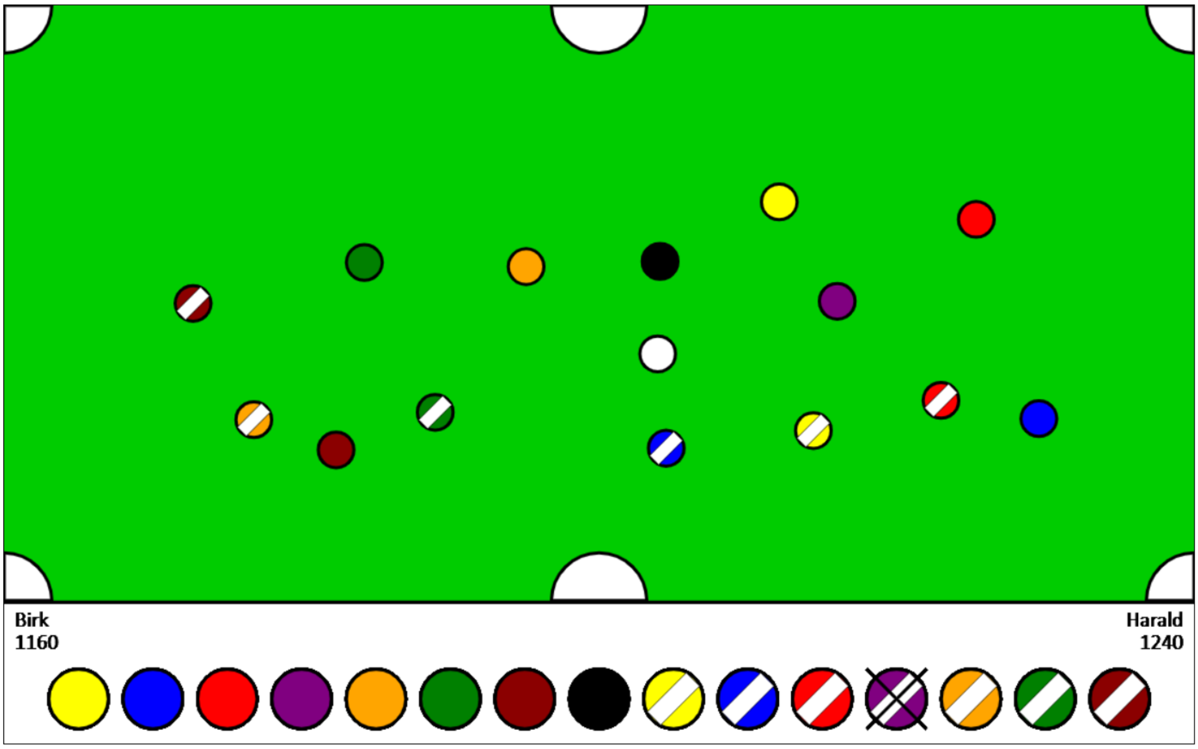


Figure 0-14: The pool table, and match information, as shown on the website

Profile

Upon entering the “Profile” page, a simple login feature is shown. This login requires an e-mail, as well as a password. If the user has not created an account before this, the user can create an account by pressing the “Create a new user”, and fill in the information needed in the, now visible, modal window. To create an account, a valid e-mail, a password, and a nickname is needed. The e-mail must be unique. Both the “Login” and the “Create a new user” functionalities is shown in Figure 0-15.

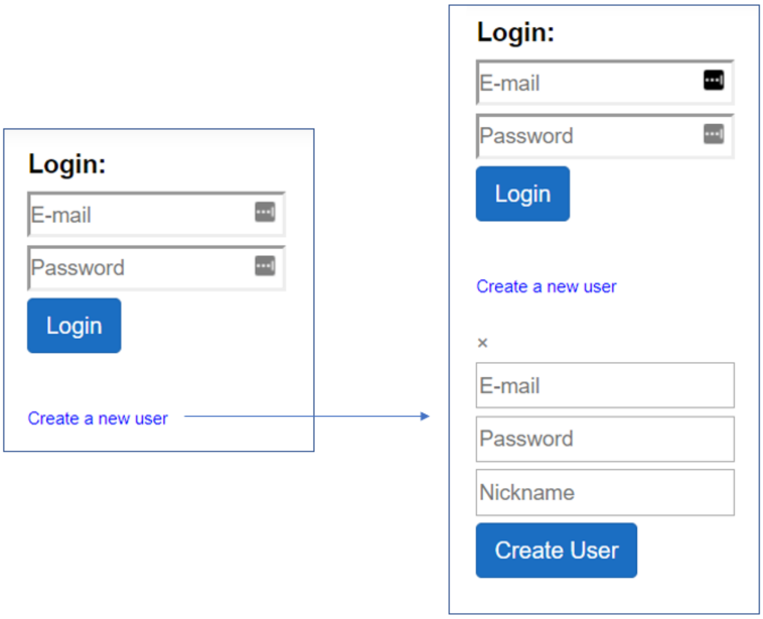


Figure 0-15: A display of the "Login", as well as the "Create a new user" feature

When logged in, there is a feature to change your nickname. This is possible after clicking the “Change nickname” text which brings up a modal window with an input for the new nickname. This is shown in Figure 0-16 below.

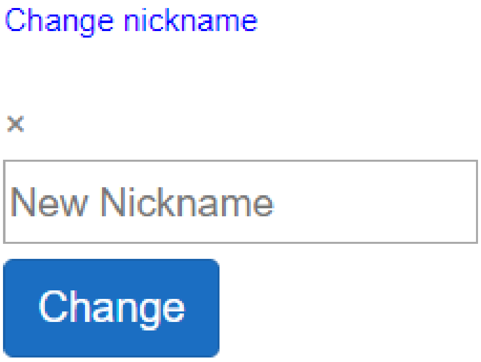


Figure 0-16: A display of the "Change nickname" modal window

Create Game

You must be logged in to use this page! How to create an account, and how to log in, is written about above, in the “Profile” chapter.

The “Create Game” contains several features:

Create a game

To create a game, you must enter the e-mail of the person you are challenging, i.e. not your own e-mail. If you do not know the other persons e-mail, you cannot play against them. After entering the e-mail, choose between different rulesets by pressing the drop-down menu. These interactions are shown in Figure 0-17. An access code will be displayed just below the “Create” button. This code is used to play the game in the LabVIEW application.

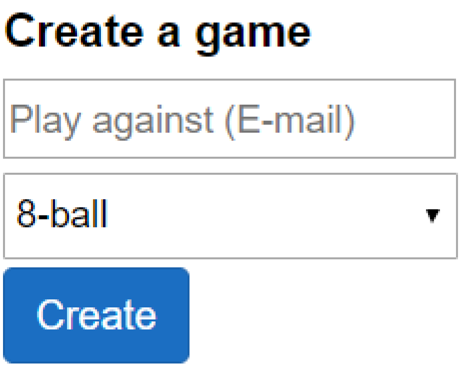


Figure 0-17: The inputs for creating a game

Join a tournament

To join a tournament, an access code is needed. This access code is given when creating a tournament, and if you have created the tournament, you will automatically be added to the tournament. To join a tournament, write the access code given by another user, who has created the tournament, into the “Access code” field. After entering the access code, press the “Join” button. This can be done as shown in Figure 0-18.

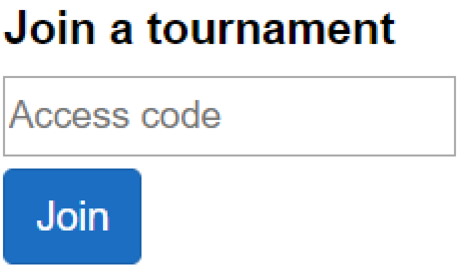


Figure 0-18: The input for joining a tournament

Creating a tournament

To create a tournament, simply press the “Create a tournament” text, then the “Create” button that appears. This returns an access code, which is needed for other users to join the tournament. This feature is displayed in Figure 0-19.

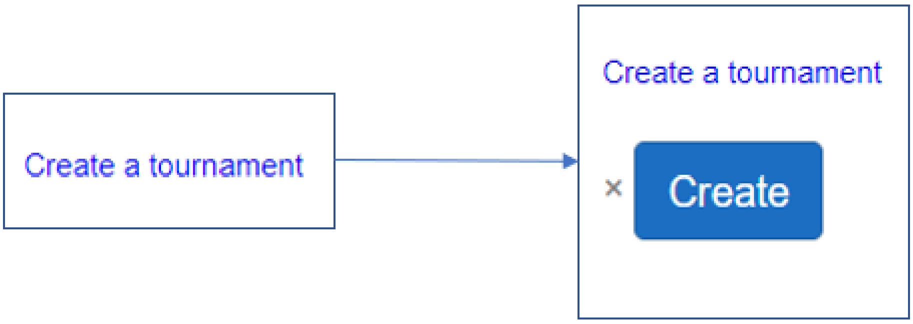


Figure 0-19: Creating a tournament, before and after pressing the "Create a tournament" text

Ranking

The “Ranking” page offers your statistics, as well as the five closest ranked users directly above, and below, you. These statistics show the different user’s nickname, MMR, total amount of wins, and the total amount of games played. To view the top 10 players, it is not needed to be logged in; however, if you wish to view your own statistics, you need to be logged in.