

SOFTWARE ENGINEERING PROCESS AND TOOLS

AUGMENTED REALITY WATCH

Date: 10 October 2020

Version	Primary Author(s)	Description of Version	Date Completed
V1.0	Serkan Akbulut	Table of Content is created	01 September 2020
V1.1	Serkan Akbulut	Overview is created	05 September 2020
V1.2	Serkan Akbulut	Overview contents are filled	10 September 2020
V2	Serkan Akbulut	Conclusion added.	10 October 2020



Project	Augmented Reality Watch	
Group Number	3	
Group Members	Sonal Cynthia Pereira	s331393
	Daljeet Singh	s332310
	Serkan Akbulut	s329315
	Aafiyaben Ahmadbhai Polra	s328620
	Imrandeep Kour	s334861
Assessment Title	Project Report	
Project GitHub Location	https://github.com/openG4NTZ/PRT582-Assessment-2	
Unit Number and Title	PRT582 Software Engineering Process and Tools	
Lecturer/Tutor	Charles Yeo	

TABLE OF CONTENTS

TABLE LIST.....	7
FIGURE LIST	8
1. INTRODUCTION.....	11
1.1. PROJECT OVERVIEW.....	11
1.2. PROJECT DELIVERABLES	11
2. PROJECT ORGANIZATION	12
2.1. SOFTWARE PROCESS MODEL	12
2.2. ROLES AND RESPONSIBILITIES.....	12
3. MARKET ANALYSIS	14
3.1. MARKET SIZE:.....	14
3.2. MARKET GROWTH RATE.....	14
3.3. MARKET PROFITABILITY:	15
3.4. INDUSTRY COST STRUCTURE:.....	15
3.5. MARKET TRENDS:.....	16
3.6. FUTURE OF AUGMENTED REALITY WATCHES:	16
4. PROJECT MANAGEMENT PLAN.....	17
4.1. DESCRIPTION.....	17
4.2. HIGH LEVEL REQUIREMENTS.....	18
4.3. WORK BREAKDOWN STRUCTURE (WBS)	18
4.4. DELIVERABLES AND MILESTONES	18
4.4.1. RESOURCES NEEDED.....	20
4.6. DEPENDENCIES	21
4.7. ASSUMPTIONS	21

4.8. CONSTRAINTS	22
4.9. RISKS AND CONTINGENCIES	22
4.9.1. PROJECT RISKS	23
4.9.2. PROJECT RISK MATRIX.....	25
4.9.3. RISK RESPONSES	26
4.10. PROJECT MANAGEMENT COMMUNICATION PLAN.....	29
5. SPECIFIC REQUIREMENTS.....	30
5.1. FUNCTIONAL REQUIREMENTS	30
5.2. NON-FUNCTIONAL REQUIREMENTS	31
5.3. SOFTWARE SYSTEM ATTRIBUTES	32
5.3.1. RELIABILITY.....	32
5.3.2. AVAILABILITY.....	32
5.3.3. SECURITY	32
5.3.4. MAINTAINABILITY	33
5.3.5. PORTABILITY	33
5.3.6. PERFORMANCE.....	33
6. SYSTEM ARCHITECTURAL DESIGN	34
6.1. DESIGN OVERVIEW.....	34
6.2. CHOSEN SYSTEM ARCHITECTURE	36
6.3. DISCUSSION OF ALTERNATIVE DESIGNS.....	38
7. AGILE DEVELOPMENT AND USER INTERFACE DESIGN	39
7.1. AGILE SPRINTS.....	40
7.1.1. SPRINT 1.....	40
7.1.2. SPRINT 2.....	40
7.1.3. SPRINT 3.....	41
7.1.4. SPRINT 4.....	41

7.1.5. SPRINT 5	42
7.1.6. SPRINT 6	43
7.1.7. SPRINT 7	44
7.1.8. SPRINT 7 AFTER COMPLETED	45
7.2. DESCRIPTION OF THE USER INTERFACE.....	46
7.2.1. OBJECTS AND ACTIONS	47
7.2.2. SCREEN IMAGES.....	48
7. TDD AND POST DEVELOPMENT TESTING	54
7.1. FEATURES TO BE TESTED	54
7.2. TEST CASES FOR TDD	55
7.2.1. TEST CASE 1 – FEATURE: OPENING THE APP.....	57
7.2.2. TEST CASE 2 – FEATURE: HOME SCREEN	59
7.2.3. TEST CASE 3 – FEATURE: WATCH CATEGORY	62
7.2.4. TEST CASE 4 – FEATURE: AR MARKER.....	64
7.2.5. TEST CASE 5 – FEATURE: LEATHER WATCH.....	66
7.2.6. TEST CASE 6 – FEATURE: SMART WATCH	67
7.2.7. TEST CASE 7 – FEATURE: METAL WATCH	68
7.2.8. TEST CASE 8 – FEATURE: SPORTS WATCH	70
7.2.9. TEST CASE 9 – FEATURE: WATCH SELECTION	71
7.2.10. TEST CASE 10 – FEATURE: AR VIEW.....	74
7.2.11. TEST CASE 11 – FEATURE: CHANGING THE WATCH	76
7.2.12. TEST CASE 12 – FEATURE: CHANGING THE COLOUR OF WATCH	78
7.2.13. OTHER TESTING SAMPLES	81
7.3. CODE REFACTORING	85
7.3.1. CODE SMELL 1: DUPLICATED CODE	85
7.3.2. CODE SMELL 2: REMOVE DUBLICATION.....	87

7.3.3. CODE SMELL 3: UNCLEAR FUNCTION NAME / RENAME	89
7.4. TEST CASES FOR POST DEVELOPMENT	90
7.4.1. PERFORMANCE TESTING:.....	92
7.4.2. SECURITY TESTING	103
7.4.3. USER EXPERIENCE TESTING	107
8. CONCLUSION	114
8.1. LESSON LEARNED	115
9. BIBLIOGRAPHY.....	116

TABLE LIST

Table 1: Roles and Responsibilities.....	13
Table 2: Deliverables and Milestones.....	20
Table 3: Functional Requirements.....	31
Table 4: Non-Functional Requirements.....	32

FIGURE LIST

Figure 1: Market Size of AR Applications.....	14
Figure 2: Market Growth	15
Figure 3: Future of the Market	16
Figure 4: Agile Methodology [4]	17
Figure 5: Project Management Stages	17
Figure 6: WBS.....	18
Figure 7: Design Process (Sommerville 2016, p.56)	34
Figure 8: User Interface Updates.....	35
Figure 9: Software Multi-tier Architecture	36
Figure 10: System Model	37
Figure 11: Alternative Design	38
Figure 12: Change Acceptance	39
Figure 13: Sprint 1.....	40
Figure 14: Sprint 2.....	40
Figure 15: Sprint 3.....	41
Figure 16: Sprint 4.....	41
Figure 17: Sprint 5.....	42
Figure 18: Sprint 6.....	43
Figure 19: Sprint 7.....	44
Figure 20: Completed Sprint 7.....	45
Figure 21: User Interface Reading data and Sending Commands.....	46
Figure 22: Sequence Diagram.....	47

Figure 23: Main Screen and Watch Selection Page.....	48
Figure 24: AR Code and Wrist Allocation.....	49
Figure 25: Leather and Smart Watches	50
Figure 26: Metal and Sports Watches	51
Figure 27: Color Change from Blue to Red on the Same Watch	52
Figure 28: Color Change on Different Watches.....	53
Figure 29: Game objects are created	58
Figure 30: Application Start	59
Figure 31	81
Figure 32	82
Figure 33	82
Figure 34	83
Figure 35	83
Figure 36	84
Figure 37	84
Figure 38	85
Figure 39: Application is Installed.....	93
Figure 40: Application Start up	94
Figure 41: Battery Level.....	96
Figure 42: Long Term Run.....	97
Figure 43: Ram Consumption Before Application and After Application is Opened	99
Figure 44: Application Size.....	100
Figure 45: Different Mobile Phones	101
Figure 46: Testing with other application on background	102

Figure 47: Key Management.....	104
Figure 48: AR_Targets.....	104
Figure 49: AR Marker.....	105
Figure 50: Secure Communication	106
Figure 51: Previous Design	108
Figure 52: After Corrective Design	109
Figure 53: Moderated Test	110
Figure 54: Right-handed Person Could Not Reach the Back Arrow Easily.	111
Figure 55: New Location of Back Button	112
Figure 56: 24 Hours Test.....	113
Figure 57: Future Enhancement	115

1. INTRODUCTION

Augmented reality (AR) is a word where users can interact with artificial computer based virtual objects where the objects that reside in the real world are enhanced sensors such as cameras or audios. AR has three basic features such as real and virtual worlds, real-time interaction between that two worlds, and accurate 3D registration to virtual objects (Wikipedia).

COVID19 has changed our shopping experience drastically. Consumers are shopping with great awareness of the environment, health and cost. This has led to huge rise in digital commerce, especially among new or low frequency consumers and this is likely to continue even after post-pandemic. So, we have developed this application which will give the consumer a different level of shopping experience where they can interact with the watch in augmented space.

Imagine that you can buy any items without visiting any store. It is questionable the social impacts of AR to community, but it is the new trend.

1.1. PROJECT OVERVIEW

This application allows the user to first choose type/category of watch. Next they can choose watch available in the gallery and can view the selected watch in augmented space. Also they can try changing the colour of the watch if it has its respective colour variants.

Agile project management approach has been used to develop the design to meet the project objectives such as customer satisfaction, better quality, designed for current, incremental, cooperative, and adaptive (Abrahamsson et al., 2017). Performance of the entire project has been continuously monitored and measured with preset milestones and key performance indicators (KPI). Any variation in the performance has been recorded and corrected accordingly.

1.2. PROJECT DELIVERABLES

System should be delivered within the client requirements. The final product will be delivered before the November 2020, satisfying the client requirements, and aligning with the project scope. User

interface design to attract the client and easy of the use for the user. English language is used for the application. Main deliverables are;

- Project report,
- Project presentation,
- Android application.

2. PROJECT ORGANIZATION

The project has been organized in such a way that all the project team members can properly communicate and coordinate well to implement the project objective. The project lifecycle has been classified in the four phases with various milestones.

2.1. SOFTWARE PROCESS MODEL

Agile project management approach has been used to develop the design as this method is driven by descriptions of customer requirement and incremental delivery. Client and other stakeholders were always available during the development of the project. Their feedback has been addressed properly during the development.

2.2. ROLES AND RESPONSIBILITIES

The major roles and responsibilities of the project team members to develop, design and execute the project are as follows:

Team Member	Roles and Responsibilities
Serkan Akbulut	<ul style="list-style-type: none">• Project management plan,• To determine project testing approach, testing tools and

	<p>environment</p> <ul style="list-style-type: none"> ● To design the testing procedure and result criteria ● To assess project dependencies and constraints ● To determine project risks and contingencies
Daljeet Singh	<ul style="list-style-type: none"> ● To develop the project overview, product overview and project deliverables ● To develop security testing ● To determine the roles and responsibilities of team members ● To design the tools and techniques of project
Aafiyaben Ahmadbhai Polra	<ul style="list-style-type: none"> ● To determine specific project requirements ● To develop user and software interfaces ● To develop communication protocols ● To design and develop high level project requirements
Sonal Cynthia Pereira	<ul style="list-style-type: none"> ● To design system architecture ● To develop user and software interfaces ● To describe the system interface and components ● To design the user interface ● To determine the features to be tested
Imrandeep Kour	<ul style="list-style-type: none"> ● Market analysis, ● To overview the entire system ● To set up deliverables and milestones ● To determine the required resources ● To assign tasks and timetable

Table 1: Roles and Responsibilities

3. MARKET ANALYSIS

The factors that are needed to be considered while analyzing augmented reality products in market are:

3.1. MARKET SIZE:

The size of market can be assessed dependent on present deals step by step information can be utilized to foresee the future deals. Presently these enlarged reality watches are made by Apple and Google so their market size is expanding. The compound yearly development rate for enlarged reality market size is expanding colossally for past years as contrast with computer generated reality worldwide market size. The AR worldwide market size increments from \$4 billion to \$161 billion in year 2022



Figure 1: Market Size of AR Applications

3.2. MARKET GROWTH RATE

The sales of previous year can be taken and trends analysis for the future sales. By this percentage growth of market can be calculated. The augmented reality market has the potential to grow by USD

76.99 billion during 2020-2024, and the market's growth momentum will decelerate during the second half of the forecast period because of the decline in year-over-year growth.



Figure 2: Market Growth

3.3. MARKET PROFITABILITY:

There is a huge benefit of augmented reality retail as from recent surveys of 1000 shoppers, conducted by e-commerce agency they found that 45% of people would be more inclined to spend larger amounts online if the technology was available that could help them envisage products before they purchased. So as with our application that would allow customers to place virtual images of the products over a real world view how it looks in their hand before buy them, it will become far easy to reach people's heart and mind and it will help in engaging more customers.

3.4. INDUSTRY COST STRUCTURE:

Costs vary greatly based on different app types: it ranges from \$5,000–\$10,000 for a simple demo app developed in 160 working hours to \$300,000+ for a feature-rich, custom-built app that takes nine months and longer to develop.

Our app is simple with higher number of features than the MVP with the aim of satisfying user's needs wants. This kind of AR app can be used as a configure or catalog, including the following features:

- Choose an item

- Visualize item on your hand
- Change the bands and dial

3.5. MARKET TRENDS:

The wearable technology market was worth USD 21.72 billion in 2016 and expected to reach USD 51.62 billion in 2022. Market growth could be attributed to the demanding lifestyle in developing countries as well as increasing advancements in wearable products such as wristwatches, smart goggles and many others.

Attractive Opportunities in the Wearable Technology Market

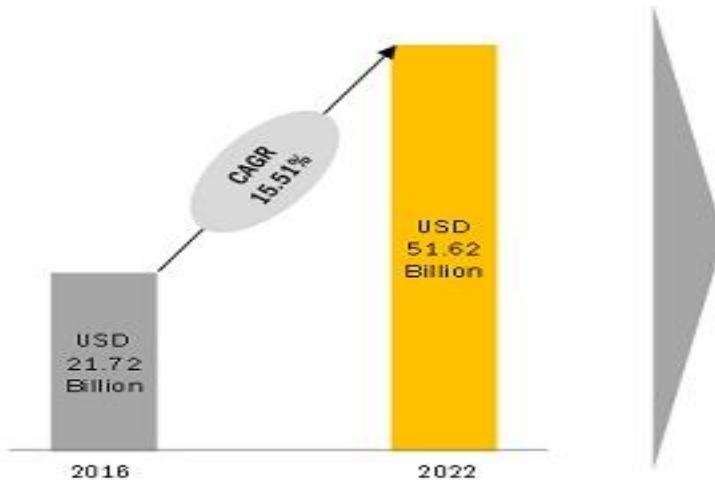


Figure 3: Future of the Market

3.6. FUTURE OF AUGMENTED REALITY WATCHES:

Specialists anticipate the AR/VR industry to arrive at more than \$25B by 2025—and the development will proceed consistently and retail area will increment by 1.6 billion. That's the brilliant fate of expanded reality, and it will be characterized by the ventures from the accompanying industry areas and circles, which locate its viable potential entirely luring.

4. PROJECT MANAGEMENT PLAN

4.1. DESCRIPTION

In this project agile development is used. From beginning of the project to until the end client fully involved with the project and client request is applied according to change management plan.

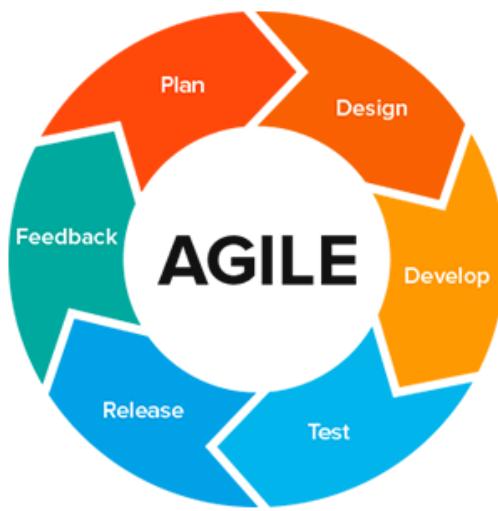


Figure 4: Agile Methodology [4]

Below figure explains project management stages in this project. In this figure development of software agile and testing and security part is the final tests. Development stages has follows above testing procedure.

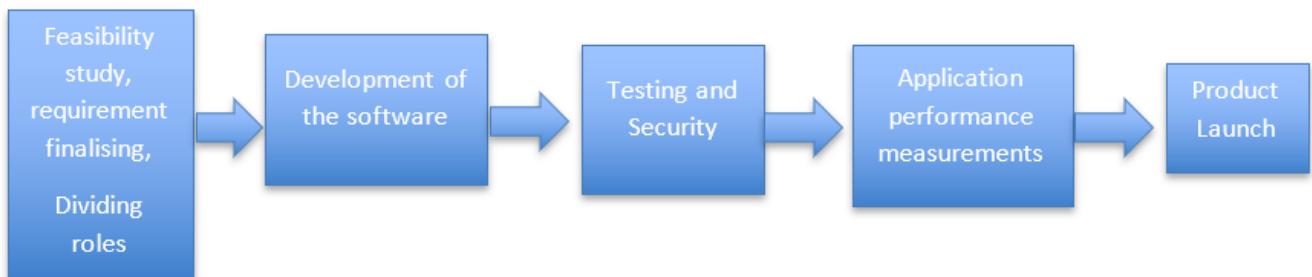


Figure 5: Project Management Stages

4.2. HIGH LEVEL REQUIREMENTS

High level project requirements of the AR Watch are:

1. AR Watch application should run on Android and IOS platforms,
2. Users should be able to use AR Watch application to virtualize the chosen watch on their arm by using AR code band,
3. Users should be able to change the colour of chosen watch,
4. AR Watch application should provide variety of straps for watches.

4.3. WORK BREAKDOWN STRUCTURE (WBS)

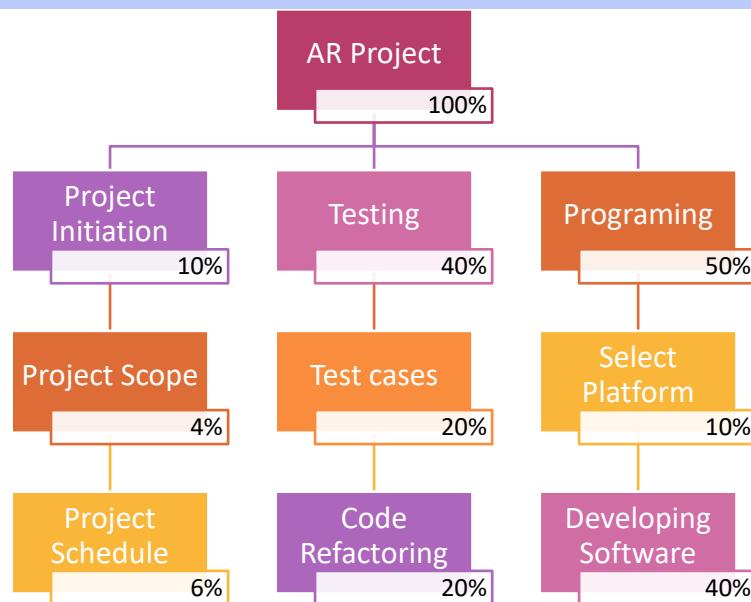


Figure 6: WBS

4.4. DELIVERABLES AND MILESTONES

The below tables consist of the milestones and the deliverables to track the progress of the project. This helps to prevent errors and complete within the time limit.

Milestones	Description	Projected time (D/M/Y)	Actual Time (D/M/Y)
Project Initiation	The project scope, objectives and stakeholders are identified. And the group contract will be sign by all members	11/08/2020	11/08/2020
Project Planning	Requirements are finalized with the user and Jira software is used to track the progress of the project	18/08/2020	18/08/2020
Prototype development	Prototype is developed and presented to the client	25/08/2020	25/08/2020
Design	MVC architecture, UML diagrams and architecture patterns are design	01/08/2020	01/08/2020
Software Coding	Software coding is done using visual studio-C#.net and Unity	23/08/2020	01/10/2020
Software Testing and Security	Functional testing will be use for the software following black box methodology. Penetration and testing injection testing for security	23/08/2020	10/10/2020
Product Launch	Completing above phases final is the product launch	01/10/2020	10/10/2020

Table 2: Deliverables and Milestones

4.4.1. RESOURCES NEEDED

AR Watch is the application where user gets the virtualization of different type of watches through smart phone application. This consists of hardware and software resources.

Development requirements:

Hardware

Different hardware that have been used to develop (ARW) are:

- Computer (Windows 10, minimum intel I3, 6 GB Ram)
- Mobile smart phone

Software

Different software that have been used to develop (ARW) are:

- Unity Software 2020.1.2f1
- Visual Studio 2019 or newer

User requirements to use application:

Hardware

- Mobile smart phone,
- AR code image to identify the origin of watch on arm.

Software

- Installed AR mobile application.

4.6. DEPENDENCIES

Dependencies are things that depends on external and internal environment. For instance, project requirements has to finish on time and it has to fulfill client requirements and changes. The client needs and changes have been approve in each phase of the project for that we conducted client meetings every week and we kept record of it with client weekly meeting reports.

Internal Dependencies

Project internal dependencies activities will be developing the code, verifying and validating by doing tests after changes, when the coding is done each change it is approved by client and new sprint started as it is a agile development. There should be no errors in functional and other testings.

External Dependencies

External dependency is the relationship between project activities and activities outside the project, where project team has no control. For example, COVID 19 restrictions, adaptation to new version of Unity Software, accidents and incidents that might have impact on the project development.

4.7. ASSUMPTIONS

- Project will be completed within the time frame client has given
- AR watch will be user friendly product,
- There will be no testing errors,
- Security will be tightening with the Data privacy
- Code written in an easy way where another person could understand.
- Meet the client satisfaction and project scope
- System will be delivering only in English Language only

User can only access to the application by IOS or ANROID and desktop is not supported

4.8. CONSTRAINTS

Communication Constraints:

- Organizational structure/ project structure,
- Communication channels,
- Gossips,
- Culture,
- Language barrier,
- Filtering good communication from bad,
- Unwillingness to share information,
- Time zone,
- Not easy-to use technology.

4.9. RISKS AND CONTINGENCIES

- Data Security
- Client information can be hacked by hackers, this information includes name, password. In this project, there is no user information recorded in the application, however in the future, user might be able to buy a watch through this application, so it is risk for future development.
- Not be able to complete the project on given time
- The team members might not be able to finish project on time because of lack of experience, internal conflicts or external problems.
- Conflicts of Interest
- Project should be developed based on the deliverables and client requirements. Project team should agree to client ideas and team should not have a different perspective against client ideas.
- Well define project scope and finalize the requirements in the initial phase of the project.
- Hardware Risks

Application might damage the user smart phone.

4.9.1. PROJECT RISKS

Category	Subcategory	Risk	Description:
Organizational	Organizational Structure	R1: Unclear authority and multiple decision makers.	If responsibilities and roles are unclear, there will be multiple decision makers or tasks will wait for to be accepted.
	Project Team:	R2: One member is causing late submission and project team does not work with that member.	There are always people who does not fulfil their responsibility. However, there are root causes for a person to be act like this. Therefore, root cause has to be identified.
Operational	Estimating	R4: Cost and time forecasts are inaccurate	Cost and time estimations fail frequently because of unexpected circumstances. Therefore, risk identification and detail of scope important to estimate accurately.
		R5: Not enough project member to complete the project.	Work breakdown structure might be not developed well therefore estimated human resources might be wrongly planned
	Planning	R6: Failed skill assessment and bad tasks allocation.	Quality of product related with skill level of employees. Poor skill assessment will lead wrong task allocation and thus will reduce the quality of project.
	Controlling	R7: Scope creep inflates scope (Usmani F 2019).	Uncontrolled changes and continuous growth of scope will make difficult to deliver project on time and budget.
	Communication	R8: Miscommunication between team members or miscommunication with client.	Team members might not use communication plan to collaborate with team members and client. This will lead miscommunication

Development	Technical	R10: Interface problems	Users might not be able to login, register or application might close itself.
		R11: System freezes	System might freeze during user was using the application.
		R12: User backlash	If user does not like to user interface or the features or privacy reasons, they might tarnish the company name.
	R13: Hardware multifunctioning		Mobile application might damage the phone.
	Security	R14: Vulnerable against cyber-attacks.	Mobile application holding user information such as location, name, phone number and email. Any security problem will put in danger to user and there might be legal consequences.
	Performance and Quality	R15: Lack of a Quality management system	Without Quality management system, performance and quality of the product will be unpredicted.

4.9.2. PROJECT RISK MATRIX

		Impact				
		Very Low 1	Low 2	Medium 3	High 4	Very high 5
Likelihood	Very high 5			R1		
	High 4			R2	R15 R9 R5	R7 R8 R4
	Medium 3			R10 R6	R12 R11	R14
	Low 2					R13 R3
	Very Low 1					

Risk Legend:

1. Very High: $80 \% \leq x \leq 100\%$
2. High: $60 \% \leq x \leq 80\%$
3. Medium: $40 \% \leq x \leq 60\%$
4. Low: $20 \% \leq x \leq 40\%$
5. Very Low: $0 \% \leq x \leq 20\%$

Ranking:

		Impact				
		Very Low 1	Low 2	Medium 3	High 4	Very high 5
Likelihood	Very high 5	5	10	15	20	25
	High 4	4	8	12	16	20
	Medium 3	3	6	9	12	15
	Low 2	2	4	6	8	10
	Very Low 1	1	2	3	4	5

4.9.3. RISK RESPONSES

Risk	Impact	Strategy	Response
R1: Unclear authority and multiple decision makers.	Decision making time will increase therefore, project deadline might be postponed.	Escalation of the risk	Risk will be responsibility of project manager.
R2: One member is causing late submission and project team does not work with that member.	Unsolved conflicts between team members will fail the project.	Risk mitigation	Communication is the easiest approach to solve conflicts. First step is understanding the problem. If problem cannot be solved within the group problem maker will be transferred to another group.
R3: Teacher fail to support project:	Teacher support is fundamental to project success. When this doesn't materialize the project fails.	Escalation of the risk	Risk will be responsibility of teacher.
R4: Cost and time forecasts are inaccurate	In accurate cost and time estimates will fail the project.	Risk mitigation	Make sure that cost and time estimates are reviewed by internal and external stakeholders and officially approved by them. If scope is too detailed, expert judgement of Delphi method will be used.
R5: Not enough project member to complete the project.	Project team will be overworked, it will reduce staff motivation. As a result of this project will fail or will not finish on schedule.	Risk mitigation	Detail of the project scope <u>has to</u> be deep enough to not miss a task in WBS. Therefore, project requirements <u>has to</u> be defined with business partners, project team, client and end users. After requirements are well defined, <u>work load</u> of the project will be defined with WBS and it will be compared with project team skill matrix. If more human resources needed, HR will hire required human resources.

R6: Failed skill assessment and bad tasks allocation.	Poorly done skill assessment will lead wrong task allocation and thus will reduce the quality of project.	Risk mitigation	Skill assessment will be done by monthly and yearly exams. Project team assessments will also be included in individual skill report and every <u>members</u> potential will be correctly analyzed and will be updated regularly. On the other hand, task requirements will be clearly explained to employee before it is assigned. This will eliminate misunderstanding of task and bad task allocation.
R7: Scope creep increases scope (<u>Usmani F 2019</u>).	Uncontrolled changes and continuous growth of scope will make difficult to deliver project on time and budget.	Risk avoidance	<p>First step, documentation of project requirements. Every <u>requirements</u> should have finalized and signed by project sponsor.</p> <p>Second step is change control process. It will assess the change and clearly show the impact to project.</p> <p>Thirdly, communication plan. After impact of the change is shown to sponsor, their approval of the change will be documented in the project status report.</p>
R8: Miscommunication between team members or miscommunication with client.	Miscommunication will lead double work, minor and major mistakes and conflicts. This will affect team collaboration. Therefore, project will fail.	Risk mitigation	<p>Regular overview the project status with weekly meetings and update schedule and cost plans. Involve stake holders to your weekly meetings and send regular emails to project team about status of the project.</p> <p>Daily, Kick-off meetings will also improve overall understanding of project between project team members.</p>
R9: Gold plating increases scope (<u>Usmani F 2019</u>).	The project team might add their own product features that aren't in requirements or change requests which will have impact on project schedule.	Risk avoidance	Without approve, adding new requirements in the project is prohibited. However, there might be useful ideas that might improve project quality. Therefore, project manager will keep open communication channels with team members.

R10: Vulnerable against cyber-attacks.	Mobile application holding user information such as location, name, phone number and email. Any security problem will put in danger to user and there will be legal consequences.	Risk mitigation	In project schedule, prototype will be tested against cyber-attacks with internal cyber security team. If it is required, an expert will be hired to test the system. Also, after downloading the application, user will accept that company will not be liable for any damage caused by cyber security problems. It is user's responsibility and therefore legal consequences will be avoided.
R11: Interface problems	Application functions might not work, and users might lose their interest in the application and popularity might reduce in the apple store and android store. Which means project failure.	Risk mitigation	Before realising final product, first prototype will be tested <u>a period of time</u> . If there is no problem, alpha version will be realised and after the user responses, beta version will be realised to minimize the software related problems. Also, after every update, application will be tested certain amount of time by 100 users.
R12: System freezes	Negative user feedbacks will increase application popularity will decrease.	Risk mitigation	User reviews and comments will be responded maximum in 2 business day. The problems they are facing will be solved in 3 business day.
R13: User backlash	It can happen after redesign UI. User backlash will tarnish company name and decrease number of users.	Risk mitigation	If there is any complains on software that is causing hardware malfunctioning, responsibility will be transferred to users and risk ownership clearly written in company-user agreement.
R14: Hardware malfunctioning	Application might overflow the RAM in users' phone. It might even cause damages on hardware. There will be legal consequences and company might have to replace the phone.	Risk transference	
R15: Lack of a Quality management system	Without Quality management system, performance and quality of the product will be unpredicted.	Risk avoidance	In the project, a well-defined quality management plan established. Quality assurance and quality control plans will be followed during life cycle of the project.

4.10. PROJECT MANAGEMENT COMMUNICATION PLAN

Communication Type	Description	Frequency	Format	Participants/ Distribution	Deliverable	Owner
Weekly Status Report	Email summary of project status	Weekly	Email	Project Sponsor, Team and Stakeholders	Status Report	Project Manager
Change Report	Meeting to review action register and status	At Change	In Person Or Video	Project Team	Weekly client meeting report	Project Manager
Change Meeting	Introduce and discuss changes and get feedbacks	At Change	In Person Or Video	Project Sponsor, Team, and Stakeholders	Report potential issues and delays	Project Manager
Stand-up Meeting	Present metrics and status to team and sponsor	Daily	In Person	Project Team	Discuss what happened yesterday and what will happen today with each team member Learnline	Project Manager
Project Review	Present project deliverables discuss feedbacks and next steps	At milestones and As Needed	In Person Or Video	Project Sponsor, Team and Stakeholders	Phase completion report and phase kickoff	Project Manager
Closure Meeting	Asses what worked and what did not, discuss lesson learned	At the End	In Person	Project Team	Lesson Learned and Project Records	Project Manager

5. SPECIFIC REQUIREMENTS

5.1. FUNCTIONAL REQUIREMENTS

PRIORITY	RANGE CONSIDERED (1-10 SCALE)
High Priority	8-10
Medium Priority	5-7
Low Priority	1-4

REQ. NO	REQUIREMENT	DESCRIPTION	PRIORITY	COMMENTS
FR1	Hand detection	The system must be able to detect customer's hand to try watch	9	
FR2	Watch detection on hand	Application must be able to detect virtual watch on user's hand.	10	
FR3	Change watch strap	After selecting watch, customers must be able choose different types of strap	8	
FR4	Change watch colour	After selecting watch, customers must be able choose different colour of the watch	8	
FR5	Customers must know price and name of specific watch	While trialling the watch on customers hand, they must be able to see the price and name of the watch	9	Requirement Taken off in Sprint 4

FR6	Capture screenshot of 3D watch on hand	User should be able to take a screen shot of watch on hand share it in social media.	7	Requirement Taken off in Sprint 6
-----	--	--	---	-----------------------------------

Table 3: Functional Requirements

5.2. NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements are shown below table.

REQ. NO	REQUIREMENT	DESCRIPTION	PRIORITY
NFR1	Performance - Start up	Mobile application should start in 5 second.	8
NFR2	Performance – Battery time	Application should not affect battery time more than 3% in usage of 5 minutes.	9
NFR3	Performance – Change watch strap	After 24 hours of application test run, there should not be any errors.	8
NFR4	Security – Cyber-attacks	Smartphone application must be secure against cyber-attacks and protect user login information.	9
NFR5	Performance- Memory consumption	Mobile application ram memory consumption should not be exceeded 500 MB.	7
NFR6	Performance – Application size	Application size should not exceed 200 MB.	8
NFR7	Performance – Hardware and Software variation	Application response should not differ with different hardware and operating systems.	8
NFR8	Performance – Usage with other Applications	While application is running it should not affect other application.	9

NFR9	User Experience- User Friendly	The system must have user friendly interface.	9
NFR10	Maintenance - Upgradable	The system must be easy to upgrade for new and changing requirements as per the company standard.	7
NFR11	User Experience- Graphics	Quality of image should be good.	8
NFR12	Availability- Platform independent	The app must be able to run on IOS and android devices	8
Updated NFR12	Availability- Platform independent	The app must be able to run on android devices	8

Table 4: Non-Functional Requirements

5.3. SOFTWARE SYSTEM ATTRIBUTES

5.3.1. RELIABILITY

AR Watch application will fulfill functional and non-functional requirements. In different environment condition, application response will not change in given time.

5.3.2. AVAILABILITY

Application will be available to be download from Apple Store or Google Play. User will be able to use application, as long as they installed it to their phone. Application do not require internet connection; therefore, it is available to user as long as mobile phone is charged.

5.3.3. SECURITY

Application will be resistant against cyber attacks to collect user information. In this stage, application does not hold any information related to users. However, hackers might try to steal technology of the product. Therefore, there will defenses to protect the code.

5.3.4. MAINTAINABILITY

Software upgrades will be released, at the same time in every continent. After, new version of application is released, customer still will be able to use older version 3 days. User feedbacks will be regularly used to improve the performance of software and adaptation to different environments.

5.3.5. PORTABILITY

Software will be able to be compiled in different platforms such IOS and Windows using unity. Therefore, application is a native application for both IOS and Android even though software code is same. On the other hand, code is stored in GitHub, therefore it can be moved to any system.

5.3.6. PERFORMANCE

Performance expectations are given in non-functional requirements. This is a native application therefore android platform should pass the requirements.

6. SYSTEM ARCHITECTURAL DESIGN

6.1. DESIGN OVERVIEW

In this section, the diagram on the left is used to explain design process.

First design inputs is explain with high level requirements, Then, design activities and design outputs are explained.

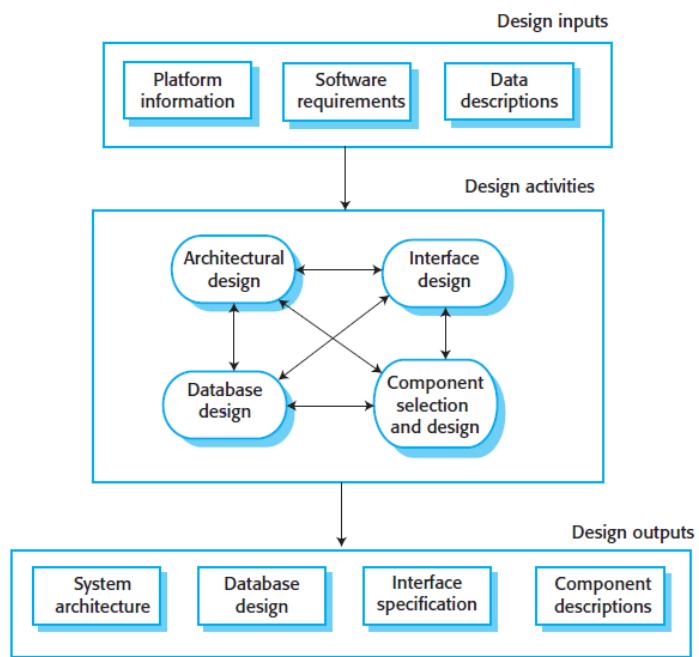
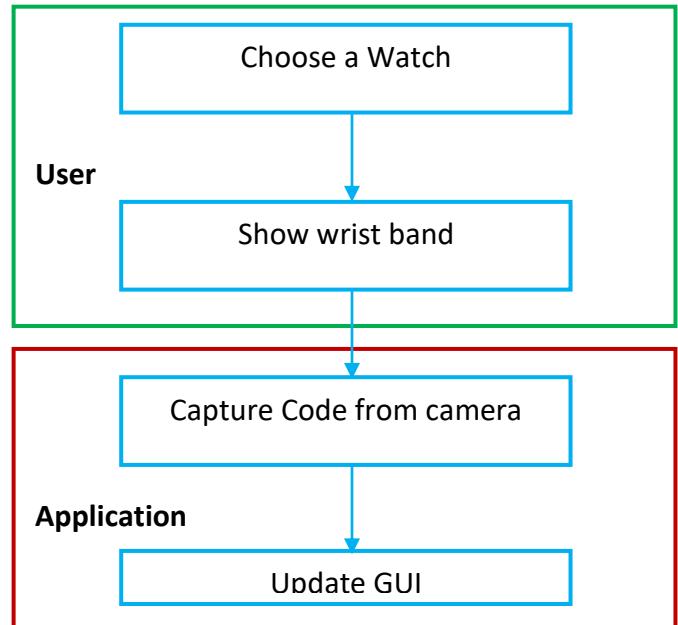


Figure 7: Design Process (Sommerville 2016, p.56)

Design Inputs

Augmented Reality Watch Application has 4 high level requirements:

- AR Watch application should run on Android and IOS platforms,
- Users should be able to use AR Watch application to virtualize the chosen watch on their arm by using AR code band,
- Users should be able to change the colour of chosen watch,
- AR Watch application should provide



variety of straps for watches.

Selected Platform:

Selected platform is Unity Software. Augment application can also be developed in Android Studio and other IOS development tools. Main reason of choosing Unity is portability. It allows developers to develop native applications in IOS and android with same code. This will reduce the development stage in the project. Version control complexity will be also reduced by half.

Data descriptions:

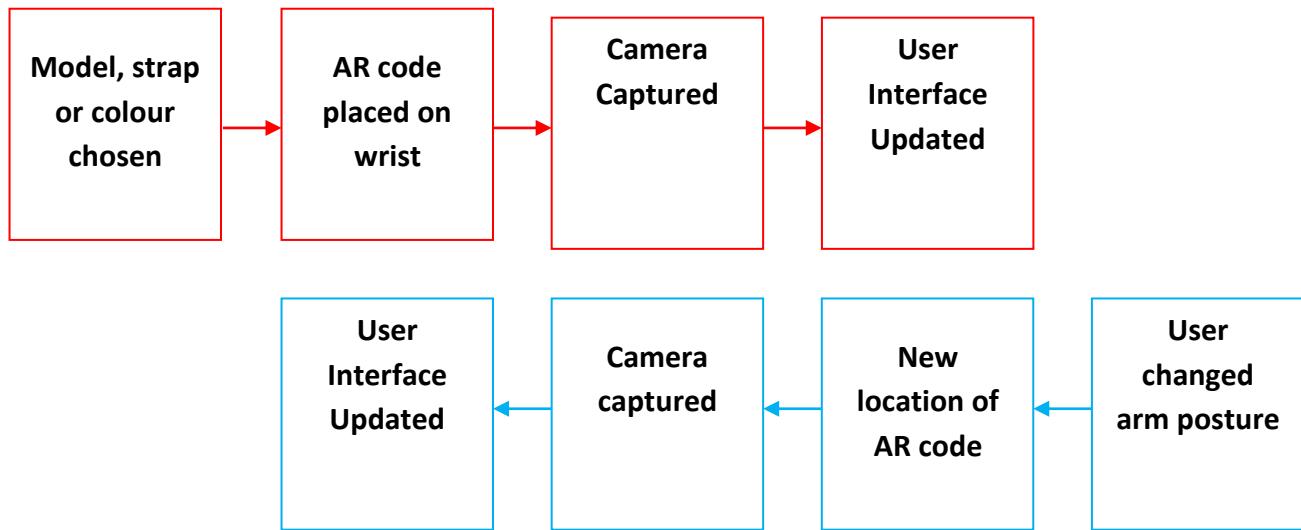


Figure 8: User Interface Updates

6.2. CHOSEN SYSTEM ARCHITECTURE

AR Watch software has 4-Tier Architecture.

First layer is application level. Application is developed in Unity version 2020.1.2f1. This software allows developers to create scripts with Visual Studio C# and create visual objects from library.

Second layer is database. Application does not have cloud based database, it uses local memory of the smart phone to save different types of watches. It has pros and cons. Good thing is it does not require internet connection to download images and therefore it is independent from internet. Thus, it has faster responses compared to cloud base databases. On the other hand, it will reduce the number of watch options because mobile phones have limited memory and applications that is larger than 200 MB is not favorable by customers.

Third layer is network layer. Application check updates via internet and in the future database might be moved to cloud servers to increase amount of watches.

Forth layer is sensor level. In this level, software interacts with smart phone camera. However, developer does not have direct access to camera. It is controlled by API's that is installed to Unity. But, in application user chooses to location of screen camera which partly related with camera peripherals.

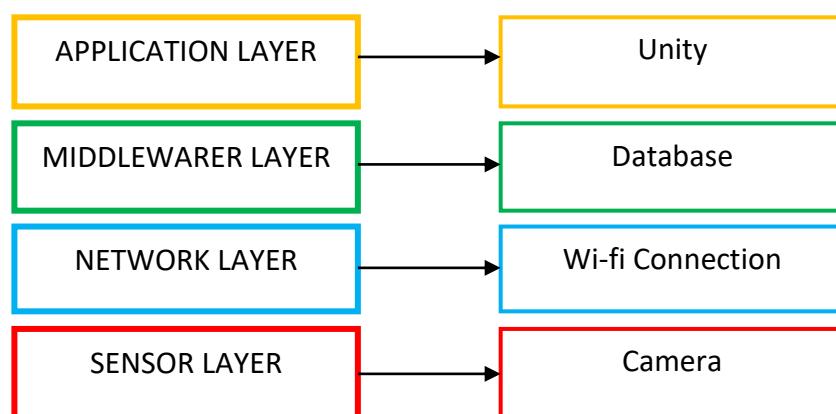


Figure 9: Software Multi-tier Architecture

User first open the application, chooses a watch, application switches to camera. Then, it scans AR code on the wrist. It displays the watch on hand.

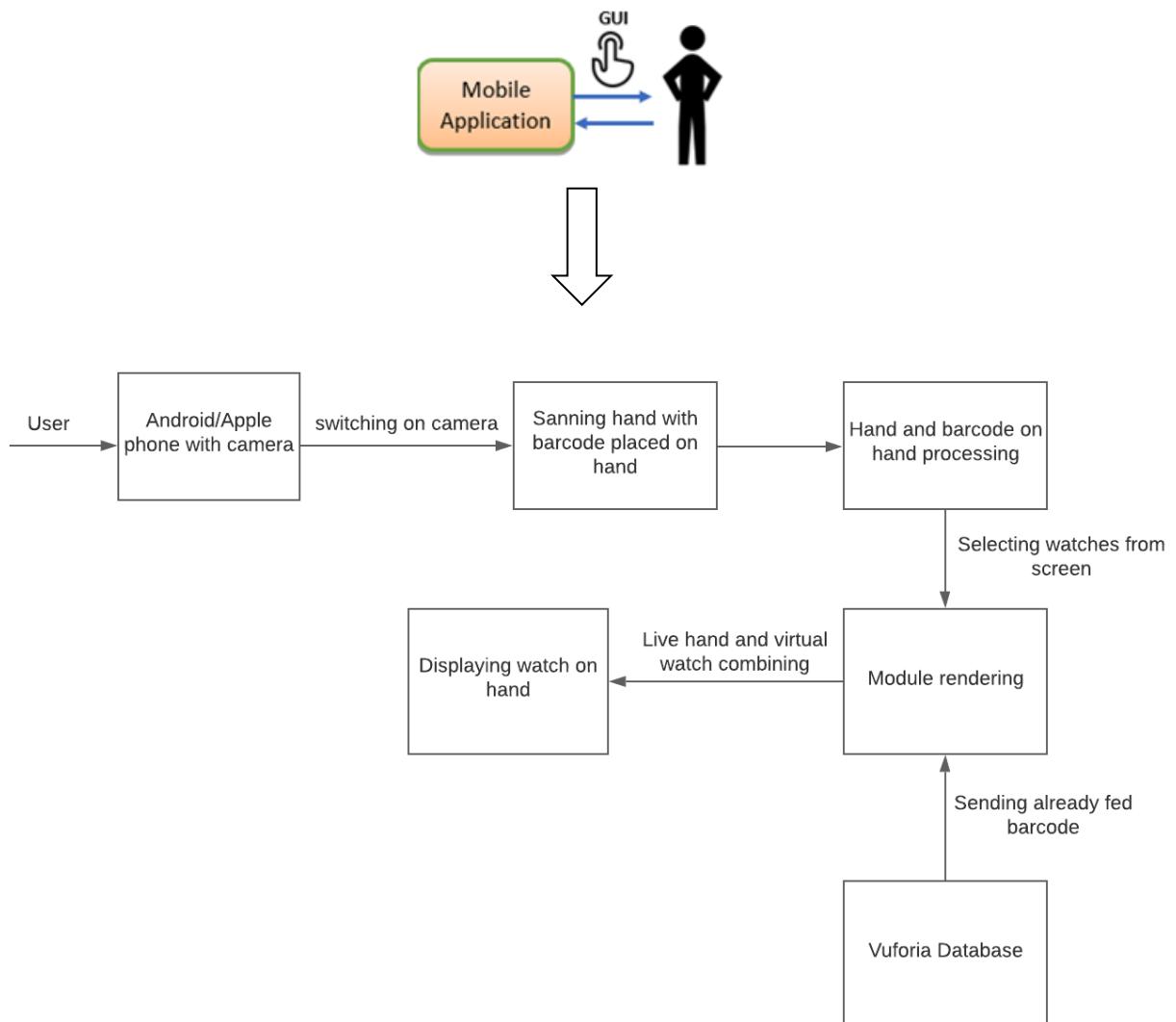


Figure 10: System Model

6.3. DISCUSSION OF ALTERNATIVE DESIGNS

Serverless architecture is an alternative type of architecture to AR watch development. In this development type, backend of the code is covered by a 3rd party company such as Azure (Microsoft) or Google Firebase. Developer only develops front end application however; a monthly fee has to be paid to 3rd party to maintenances and rent of the server. Because of complexity of project, experience requirements and server rental price application first developed without database. In the future, it can be moved to cloud.

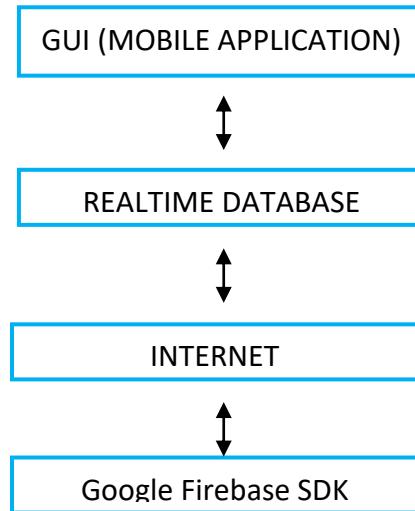


Figure 11: Alternative Design

7. AGILE DEVELOPMENT AND USER INTERFACE DESIGN

In this project, we met with client every week. Client is informed what has been done so far and client feedbacks are received to start new sprint to make requested changes. Change acceptance diagram is shown below.

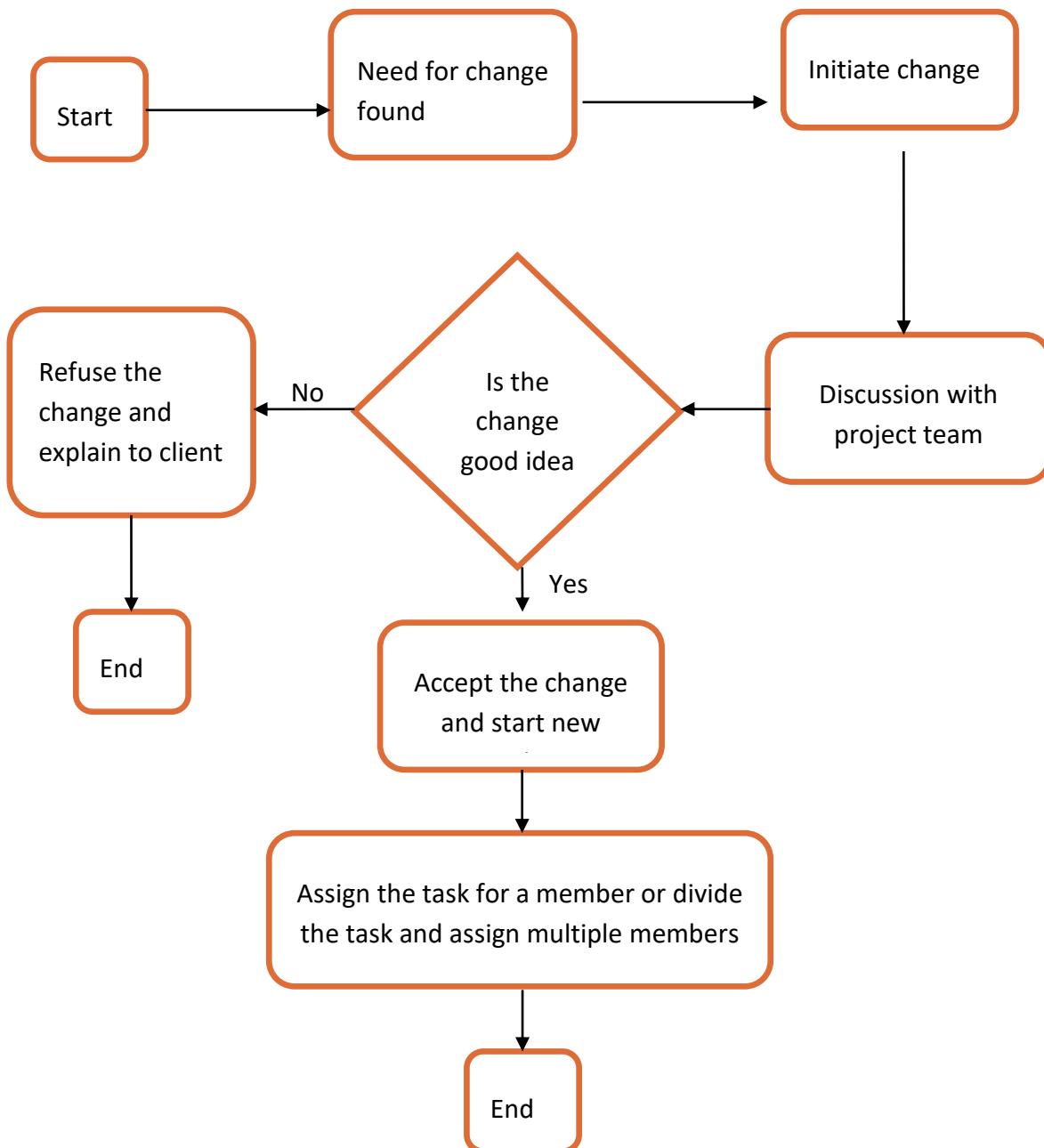


Figure 12: Change Acceptance

7.1. AGILE SPRINTS

7.1.1. SPRINT 1

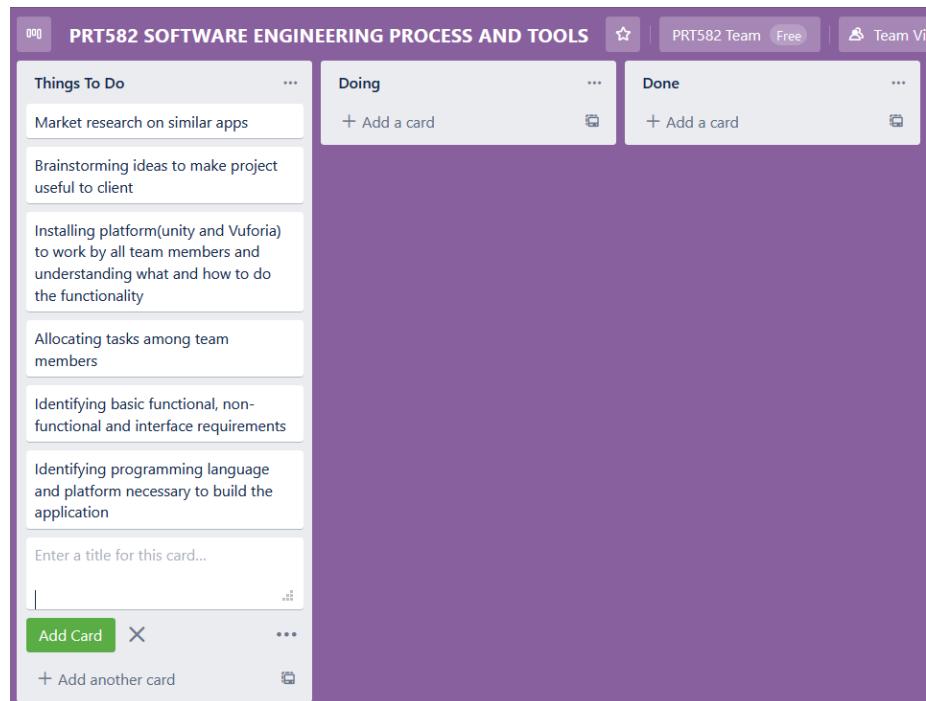


Figure 13: Sprint 1

7.1.2. SPRINT 2

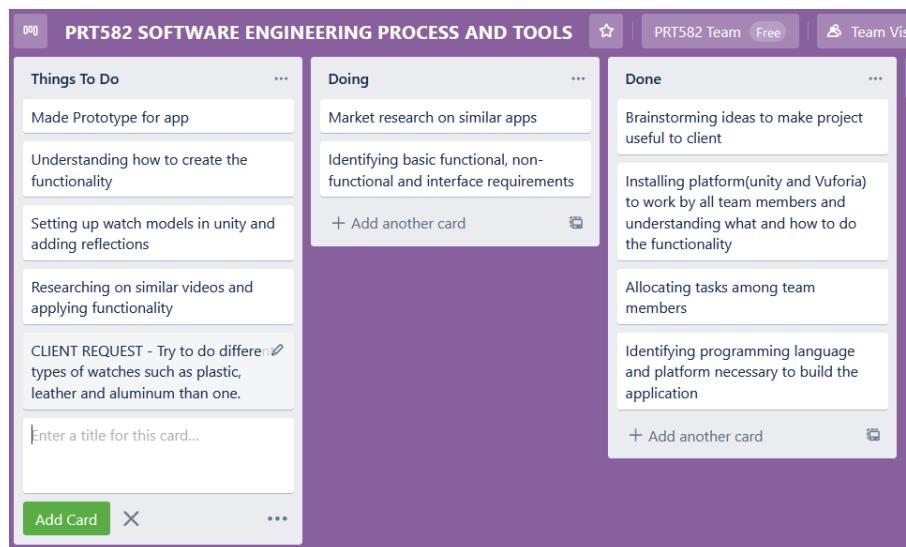


Figure 14: Sprint 2

7.1.3. SPRINT 3

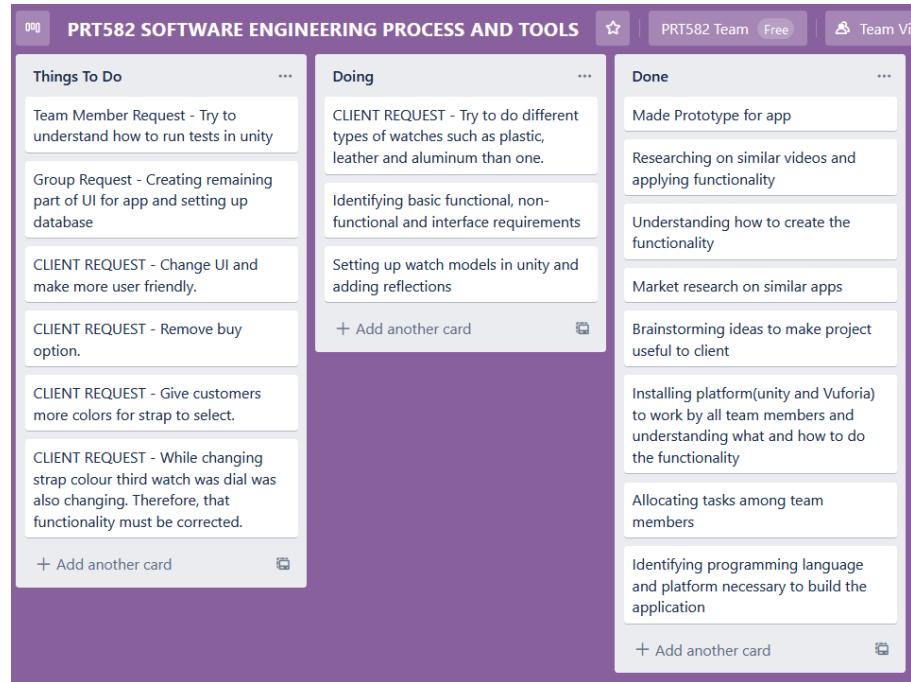


Figure 15: Sprint 3

7.1.4. SPRINT 4

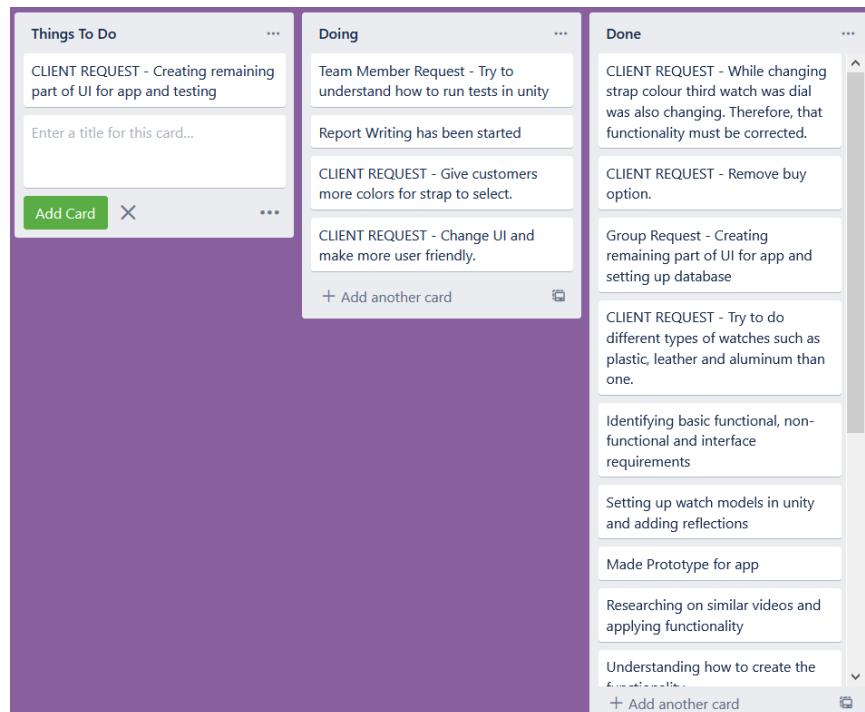


Figure 16: Sprint 4

7.1.5. SPRINT 5

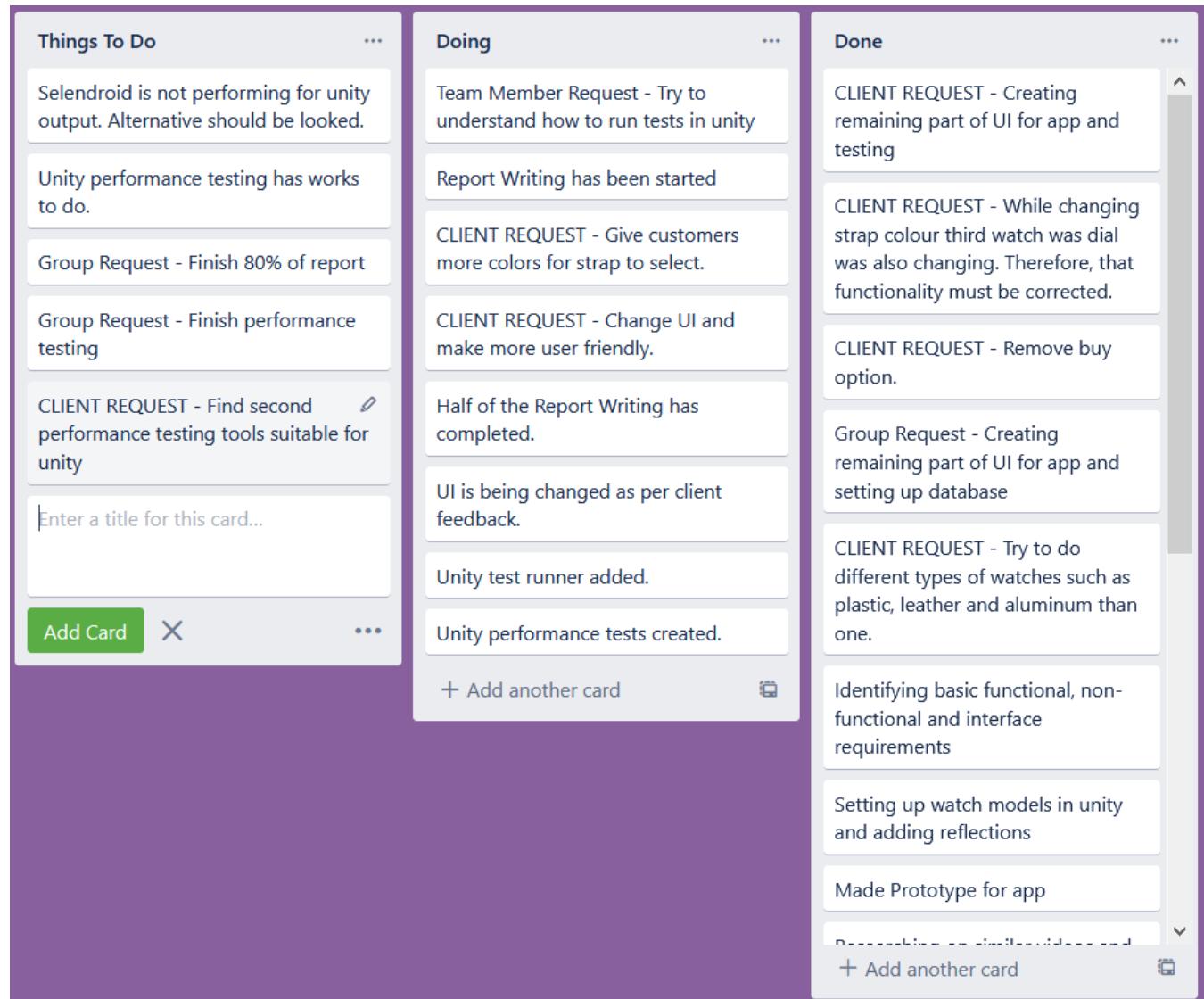


Figure 17: Sprint 5

7.1.6. SPRINT 6

The image shows a Trello board titled "PRT582 SOFTWARE ENGINEERING PROCESS AND TOOLS". The board is organized into three main columns: "Things To Do", "Doing", and "Done".

- Things To Do:**
 - Group Request-Finish AR watch application.
 - Group Request- Finish Report Writing.
 - CLIENT REQUEST - Unity social sharing and screenshots by customers.
 - CLIENT REQUEST - For final application made .APK file
- Doing:**
 - CLIENT REQUEST - Find second performance testing tools suitable for unity
 - Team Member Request - Try to understand how to run tests in unity
 - Selendroid is not performing for unity output. Alternative should be looked.
 - Group Request - Finish performance testing
- Done:**
 - Unity performance testing has works to do.
 - Unity test runner added.
 - UI is being changed as per client feedback.
 - Unity performance tests created.
 - CLIENT REQUEST - Give customers more colors for strap to select.
 - CLIENT REQUEST - Change UI and make more user friendly.
 - Group Request - Finish 80% of report
 - Report Writing has been started
 - Half of the Report Writing has completed.
 - CLIENT REQUEST - Creating remaining part of UI for app and testing
 - CLIENT REQUEST - While changing strap colour third watch was dial

At the bottom left of the board, there is a green button labeled "Add Card". At the bottom right, there is a blue button labeled "+ Add another card".

Figure 18: Sprint 6

7.1.7. SPRINT 7

PRT582 SOFTWARE ENGINEERING PROCESS AND TOOLS

Things To Do

- Group Request - Finish 100% AR watch application
- Group Request - Finish 100% of report
- Group Request - Prepare presentation
- CLIENT REQUEST - Change allocation of watches on the screen.
- CLIENT REQUEST - Change the allocation of the AR Code on the screen.
- CLIENT REQUEST - Save AR Code in local store not in the cloud and share it with a button.
- Enter a title for this card...

Doing

- Group Request-Finish AR watch application.
- Group Request- Finish Report Writing.

Done

- CLIENT REQUEST - For final application made .APK file
- CLIENT REQUEST - Unity social sharing and screenshots by customers.
- Selendroid is not performing for unity output. Alternative should be looked.
- Group Request - Finish performance testing
- Team Member Request - Try to understand how to run tests in unity
- CLIENT REQUEST - Find second performance testing tools suitable for unity
- Unity performance testing has works to do.
- Unity test runner added.
- UI is being changed as per client feedback.

Figure 19: Sprint 7

7.1.8. SPRINT 7 AFTER COMPLETED

PRT582 SOFTWARE ENGINEERING PROCESS AND TOOLS

Things To Do

+ Add a card

Doing

+ Add a card

Done

- Group Request - Finish 100% AR watch application
- Group Request - Finish 100% of report
- Group Request - Prepare presentation
- CLIENT REQUEST - Save AR Code in local store not in the cloud and share it with a button.
- CLIENT REQUEST - Change the allocation of the AR Code on the screen.
- CLIENT REQUEST - Change allocation of watches on the screen.
- Group Request-Finish AR watch application.
- Group Request- Finish Report Writing.
- CLIENT REQUEST - For final application made .APK file
- CLIENT REQUEST - Unity social

+ Add another card

Figure 20: Completed Sprint 7

7.2. DESCRIPTION OF THE USER INTERFACE

In the project, user connects to system with android mobile application. Below diagram show user interaction with application.

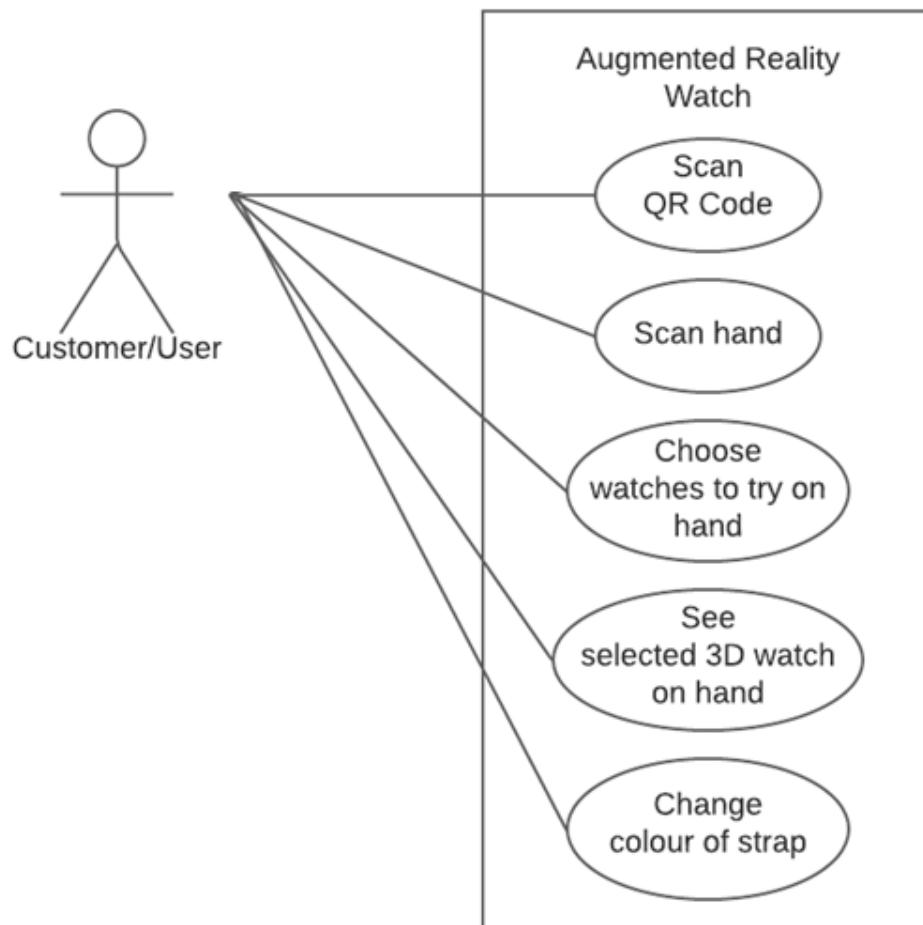


Figure 21: User Interface Reading data and Sending Commands

7.2.1. OBJECTS AND ACTIONS

Objects and actions are defined below sequence diagram.

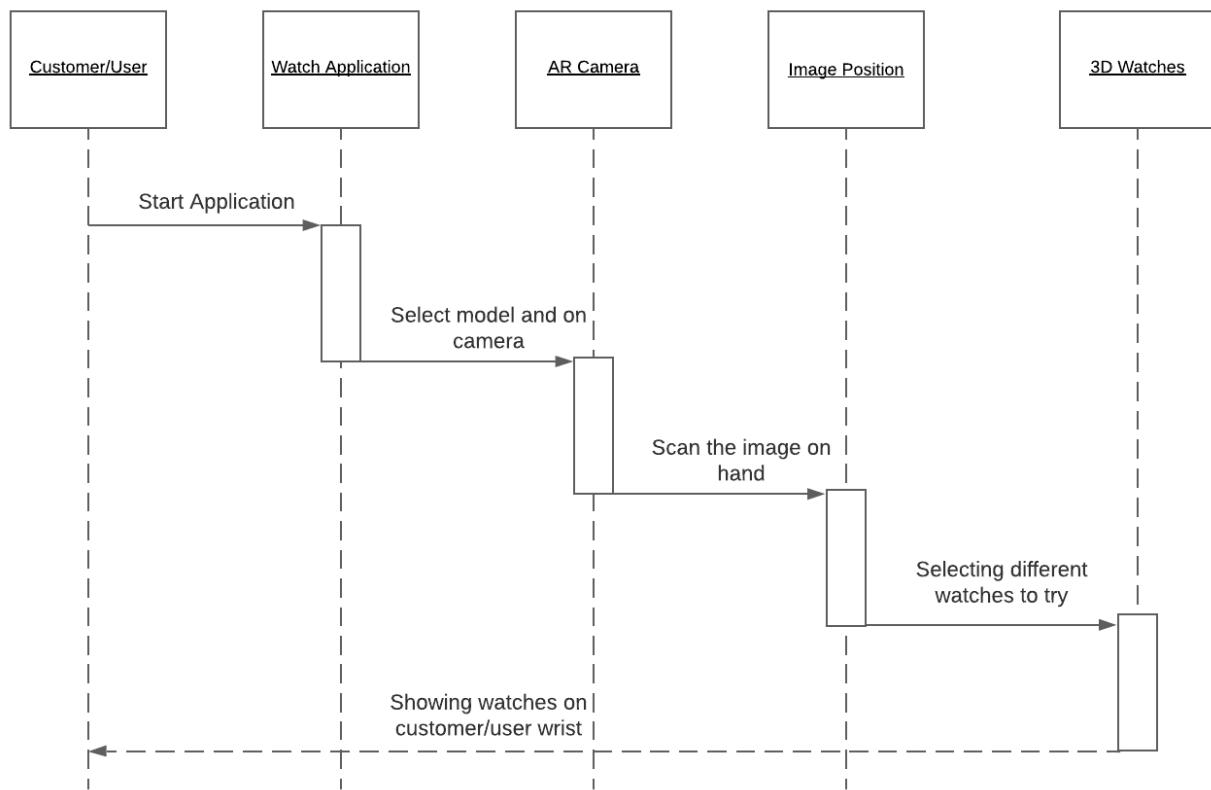


Figure 22: Sequence Diagram

7.2.2. SCREEN IMAGES

7.2.2.1. MAIN SCREEN AND WATCH SELECTION

The users of the mobile application for should see the launch page when the user open the application, see Figure 23. Using the Gif file as the background implies AR user desire for exploration. Background will show Continuously after clicking on the page that second page will be come.

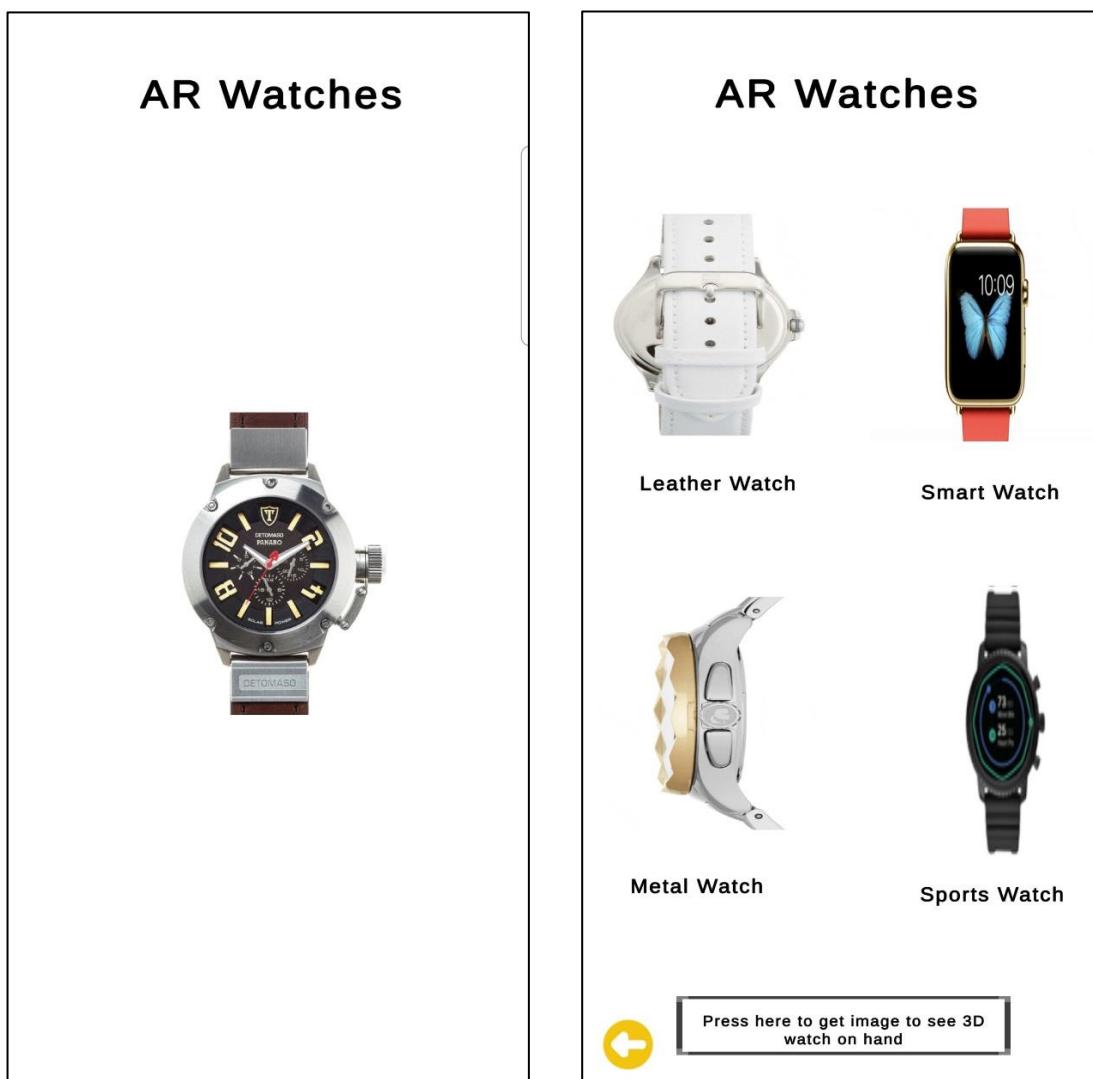


Figure 23: Main Screen and Watch Selection Page

7.2.2.2. AR CODE AND WRIST ALLOCATION

For AR marker, User need to press on the button which is shown on Figure 23 and user will get image to see 3D watch as it shown in Figure 24.

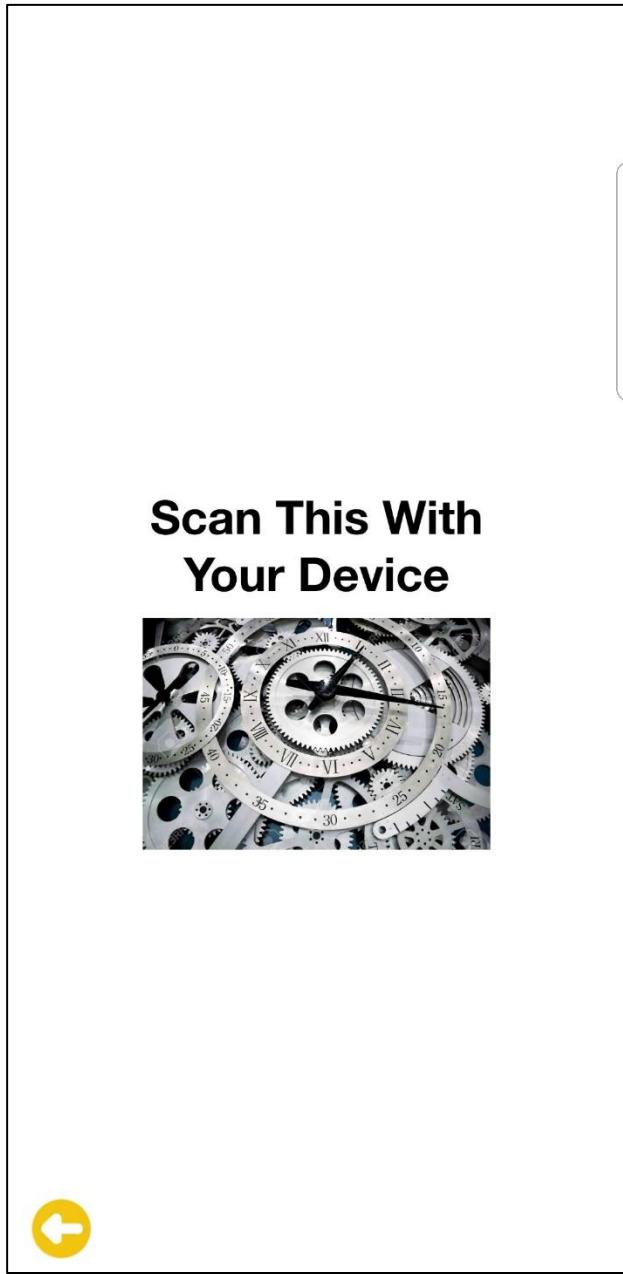


Figure 24: AR Code and Wrist Allocation

7.2.2.3. STRAP SELECTION

If the users choose one of the four categories, the next page will be shown, see Figure 25. For example, a user chooses “Lather watch”, four lather watch will be shown with watch name and company’s name.

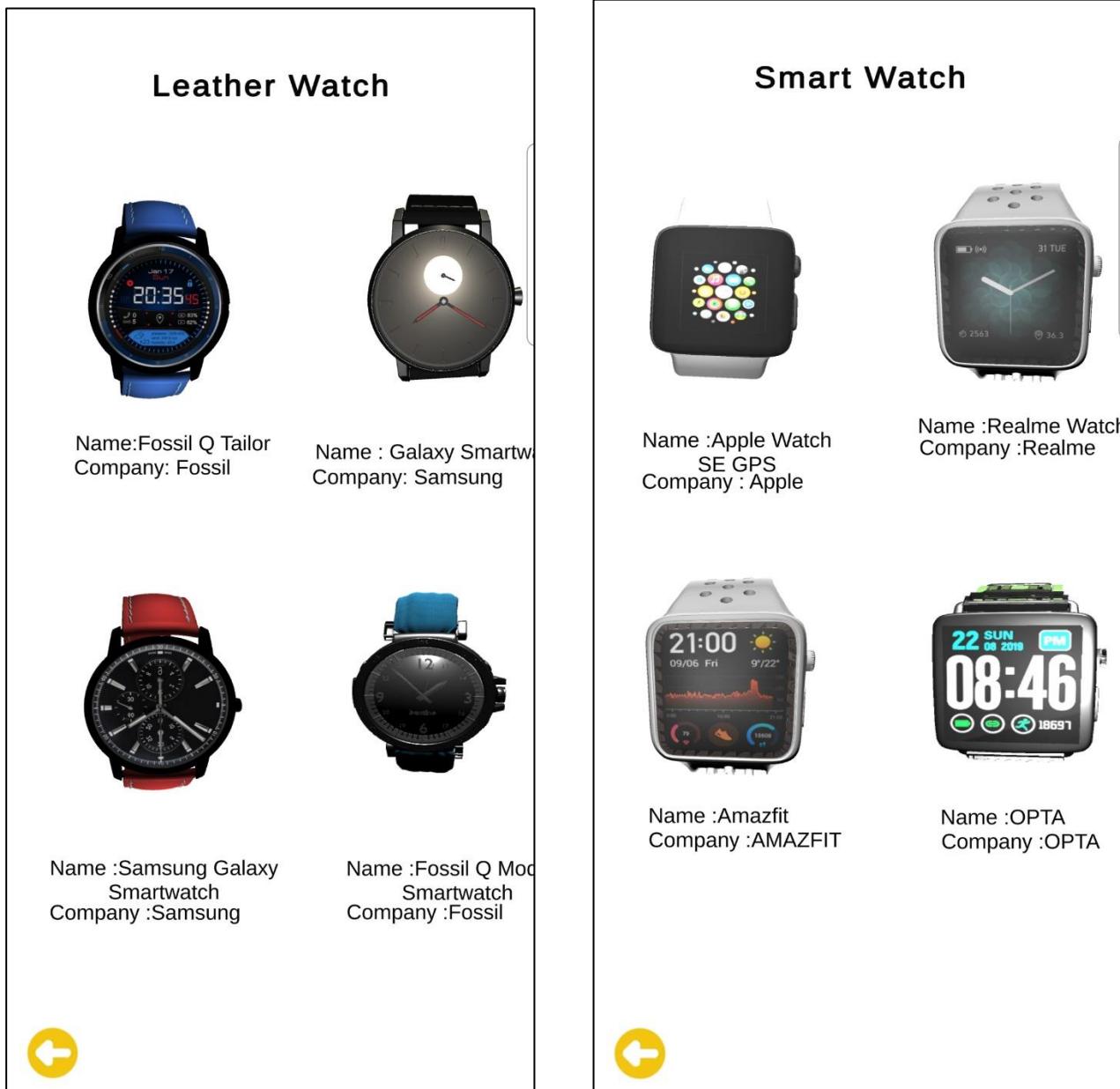


Figure 25: Leather and Smart Watches

Metal Watch



Name :DressBerry
Company :DressBerry



Name :Armani Exchange
Company :A|X



Name : Fossil
Company : Fossil



Name :TRYST
Company :TRYST



Sports Watch



Name :DressBerry 1487945
Analog Watch
Company :DressBerry



Name :Fossil
Company :Fossil



Name :Roadster
Company :Roadster



Name :Scuderia Ferrari
Company :Scuderia Ferrari



Figure 26: Metal and Sports Watches

7.2.2.4. COLOUR SELECTION



Figure 27: Color Change from Blue to Red on the Same Watch

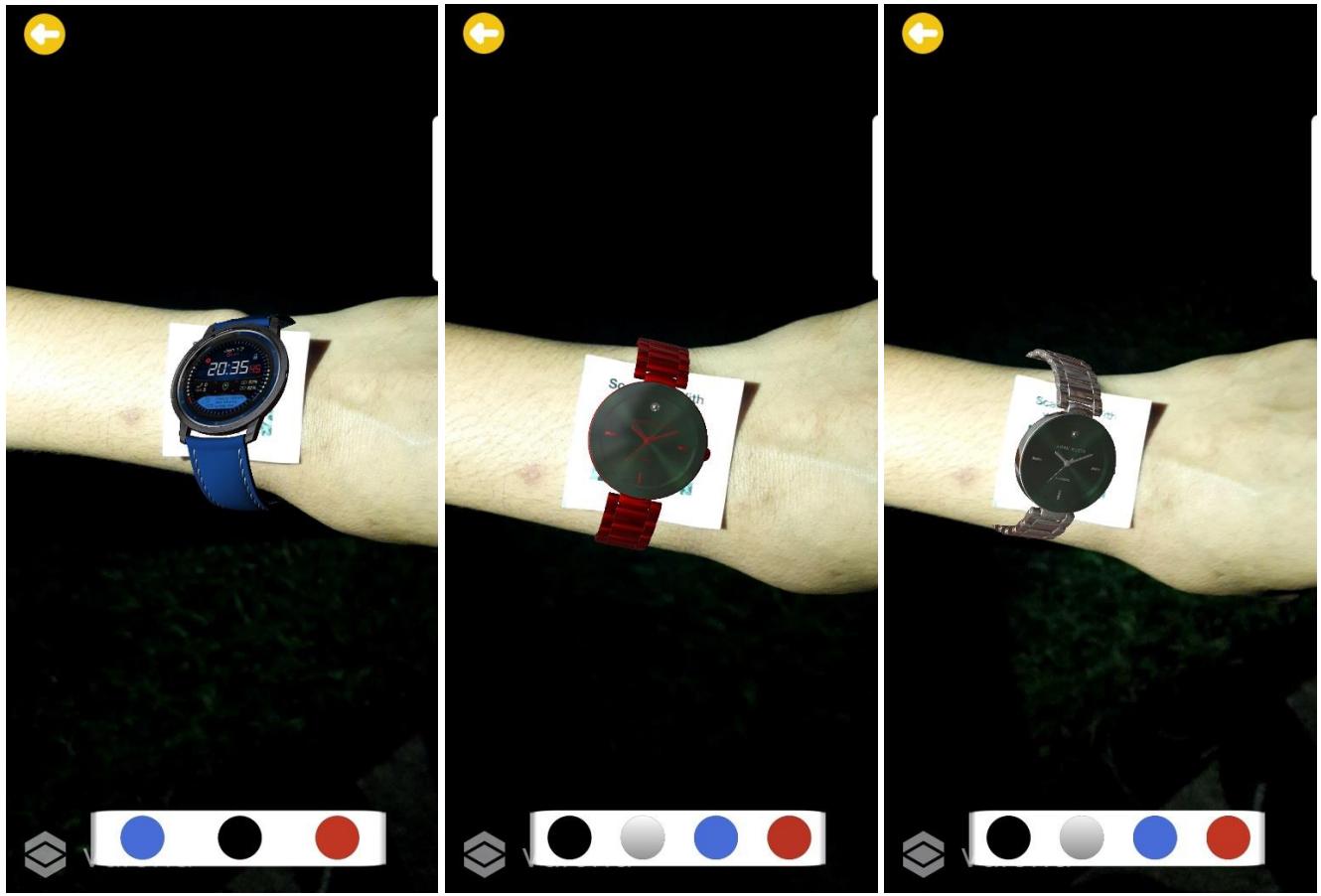


Figure 28: Color Change on Different Watches

7. TDD AND POST DEVELOPMENT TESTING

7.1. FEATURES TO BE TESTED

The Testing Traceability Matrix below, describes the features to be tested corresponding to the following categories:

- A. User Interface
- B. Maintenance
- C. Performance Testing
- D. Security Testing
- E. User Experience Testing

Testing Traceability Matrix

Traceability ID	Category	Requirement ID	Requirement Description	Test Case ID	Status
A	User Interface	FR1, FR2, FR3, FR4	User interface should pass the functional requirements	TC001, TC002, TC003, TC004, TC005, TC006, TC007, TC008, TC009, TC010, TC011	Pass
B	Maintenance	NFR10, NFR12	Application must be maintainable	TC0013	Pass
C	Performance	NFR1, NFR2, NFR3, NFR5, NFR6, NFR7, NFR8	Application must pass non-functional requirements	TC014, TC015, TC016, TC0017, TC018, TC019, TC0020,	Pass
D	Security	NFR4	Application must	TC0021	Pass

			pass security tests		
E	User Experience	NFR9, NFR11	Application must pas user experience requierements.	TC0022	Pass

7.2. TEST CASES FOR TDD

Test case ID	Feature	Preconditions	Test Procedure	Expected outputs
TC001	Opening the app		Click the app icon to open the application	For the first time Android devices will ask for “Allow the app to take picture and record video” permission
TC002	Home Screen	TC001	Click “Allow” to enter into the app	App home screen should be appear with different watches changing automatically
TC003	Watch Category	TC002	Click on the watches	User will be directed to the watch selection page
TC004	AR marker	TC003	Click on “Press here to get image to see 3D watch on hand”	User will be directed to the Marker page

TC005	Leather Watch	TC004	Click on “Leather Watch”	User will be directed to the leather watch gallery page
TC006	Smart Watch	TC004	Click on “Smart Watch”	User will be directed to the leather watch gallery page
TC007	Metal Watch	TC004	Click on “Metal Watch”	User will be directed to the Metal watch gallery page
TC008	Sports Watch	TC004	Click on “Sports Watch”	User will be directed to the Sports watch gallery page
TC009	Watch Selection	TC005/TC006/ TC007/TC008	Select the watch available in the gallery	Camera will open with the back button on the top and colour selection on the middle right side of the camera
TC010	AR View	TC009	Scan the marker to view the 3D model of the watch	The 3D model of the watch will be displayed
TC011	Changing the watch	TC010	Click back button on the left top corner	Watch gallery page will be displayed and user can select the watch or go back and change the category view a different type of watch

TC012	Changing the colour of watch	TC010	Choose the colour available on the AR view	The watch colour will be changed accordingly if available for that model
-------	------------------------------	-------	--	--

7.2.1. TEST CASE 1 – FEATURE: OPENING THE APP

FAILED: BECAUSE THERE IS NO CODE DEVELOPED

TC001	Opening the app		Click the app icon to open the application	Failed
-------	-----------------	--	--	--------

There is no code there for test is failed in the first run.

PASSED: AFTER MAIN SCREEN IS DEVELOPED

```
// Start is called before the first frame update
void Start()
{
    string page_number ;
    page_number = PlayerPrefs.GetString("namepage");
    switch (page_number)
    {
        case "mainpage":
            page1.gameObject.SetActive(true);
            page2.gameObject.SetActive(false);
            W_page1.gameObject.SetActive(false);
            W_page2.gameObject.SetActive(false);
            W_page3.gameObject.SetActive(false);
            W_page4.gameObject.SetActive(false);
            QR_page.gameObject.SetActive(false);
            break;
    }
}
```

Figure 29: Game objects are created

This case is tested every time code is run.

TC001	Opening the app	Click the app icon to open the application	Passed
-------	-----------------	--	--------



Figure 30: Application Start

7.2.2. TEST CASE 2 – FEATURE: HOME SCREEN

TC002	Home Screen	TC001	Click image to enter into the app	App home screen should be appear with different watches changing	pass
-------	-------------	-------	-----------------------------------	--	------

				automatically	
--	--	--	--	---------------	--

TESTING CODE

```
namespace Tests
{
    public class TestScript
    {
        // A Test behaves as an ordinary method
        [Test]
        public void Test MainPage()
        {
            // Use the Assert class to test conditions
            var go = new GameObject("WatchesPages");
            Assert.AreEqual("mainpage", go.start("mainpage"));
        }
    }
}
```

CODE IS DEVELOPED

```
public class WatchesPages : MonoBehaviour
{
    public GameObject page1, page2,W_page1, W_page2,W_page3, W_page4, QR_page;

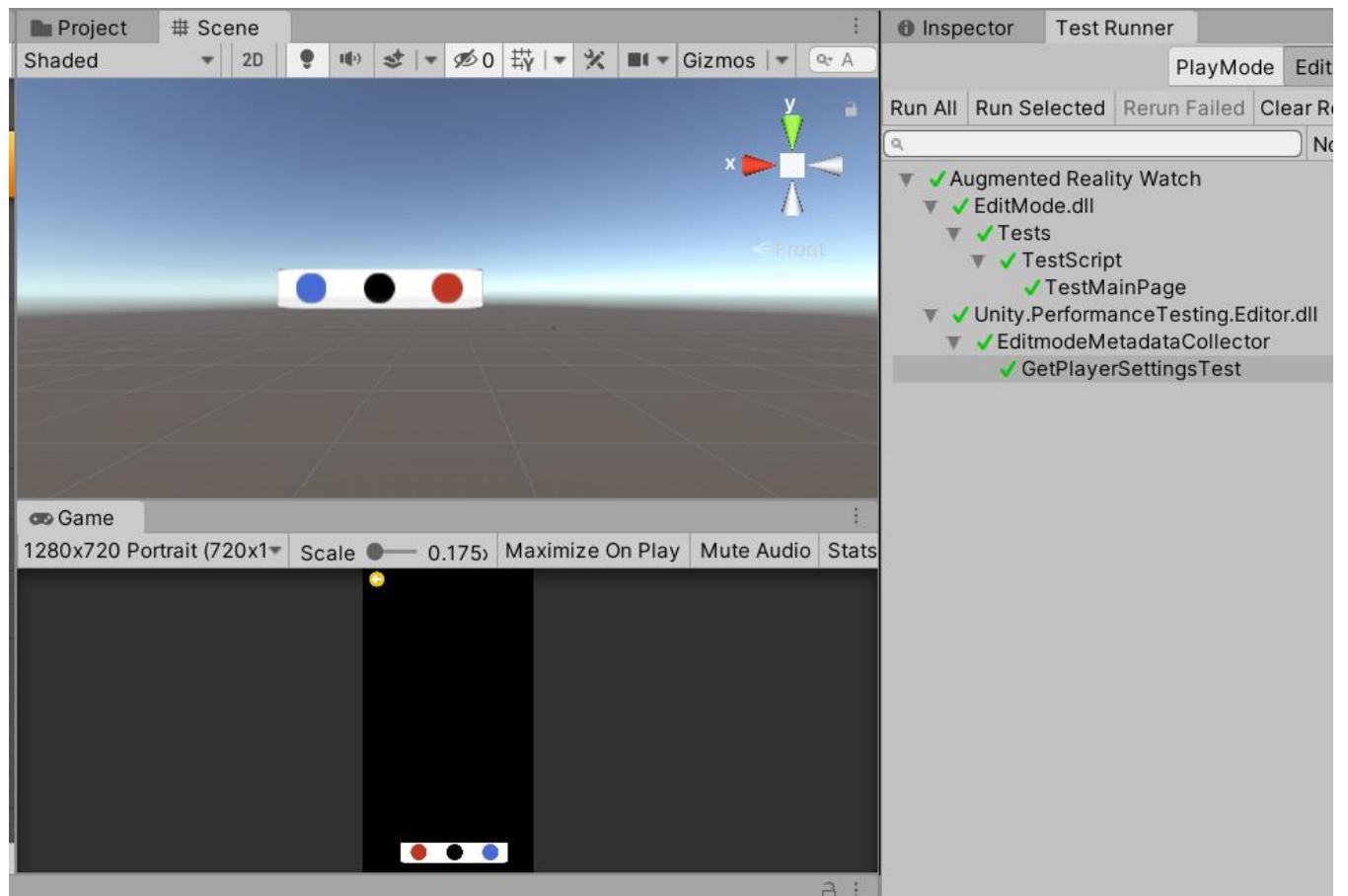
    // Start is called before the first frame update
    void Start()
    {

        string page_number ;
        page_number = PlayerPrefs.GetString("namepage");
        switch (page_number)
        {

            case "mainpage":
                page1.gameObject.SetActive(true);
                page2.gameObject.SetActive(false);
                W_page1.gameObject.SetActive(false);
                W_page2.gameObject.SetActive(false);
                W_page3.gameObject.SetActive(false);
                W_page4.gameObject.SetActive(false);
                QR_page.gameObject.SetActive(false);

                break;
        }
    }
}
```

CODE IS PASSED



7.2.3. TEST CASE 3 – FEATURE: WATCH CATEGORY

TC003	Watch Category	TC002	Click on the watches	User will be directed to the watch selection page	Passed
-------	----------------	-------	----------------------	---	--------

TESTING CODE

```
namespace Tests
{
    public class TestScript
    {
        // A Test behaves as an ordinary method
        [Test]
        public void TestMainPage()
        {
            // Use the Assert class to test conditions
            var go = new GameObject("WatchesPages");
            Assert.AreEqual("mainpage", go.start("mainpage"));
        }

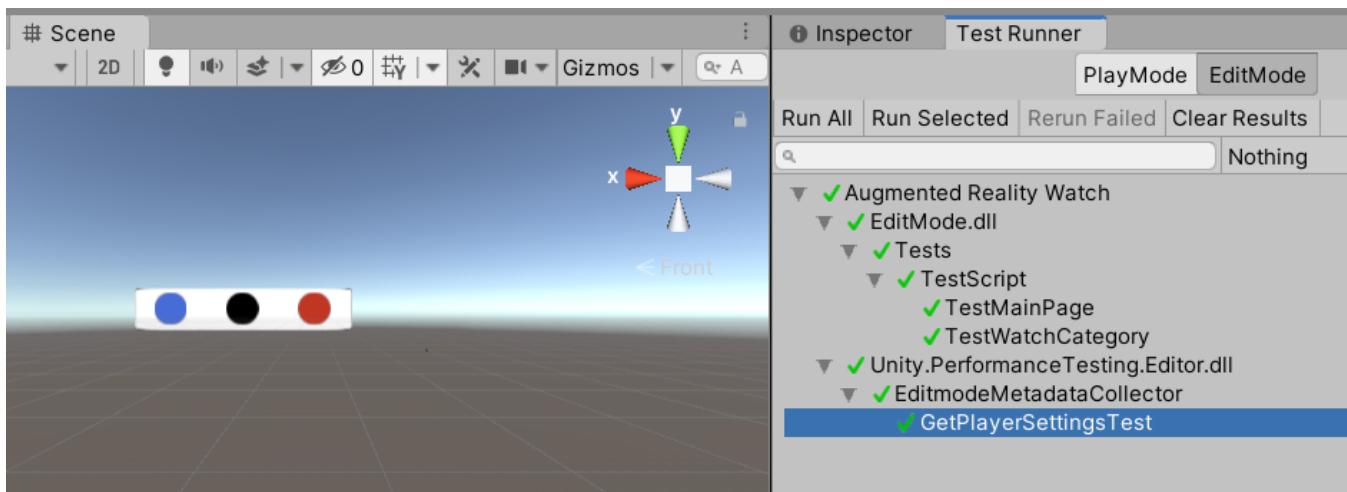
        [Test]
        public void TestWatchCategory()
        {
            // Use the Assert class to test conditions
            var go = new GameObject("WatchesPages");
            Assert.AreEqual("watchtypes", go.start("watchtypes"));
        }
    }
}
```

CODE IS DEVELOPED

```
case "watchtypes":
    page1.gameObject.SetActive(false);
    page2.gameObject.SetActive(true);
    W_page1.gameObject.SetActive(false);
    W_page2.gameObject.SetActive(false);
    W_page3.gameObject.SetActive(false);
    W_page4.gameObject.SetActive(false);
    QR_page.gameObject.SetActive(false);

    break;
```

CODE IS PASSED



7.2.4. TEST CASE 4 – FEATURE: AR MARKER

TC004	AR marker	TC003	Click on “Press here to get image to see 3D watch on hand”	User will be directed to the Marker page	Passed
-------	-----------	-------	--	--	--------

TESTING CODE

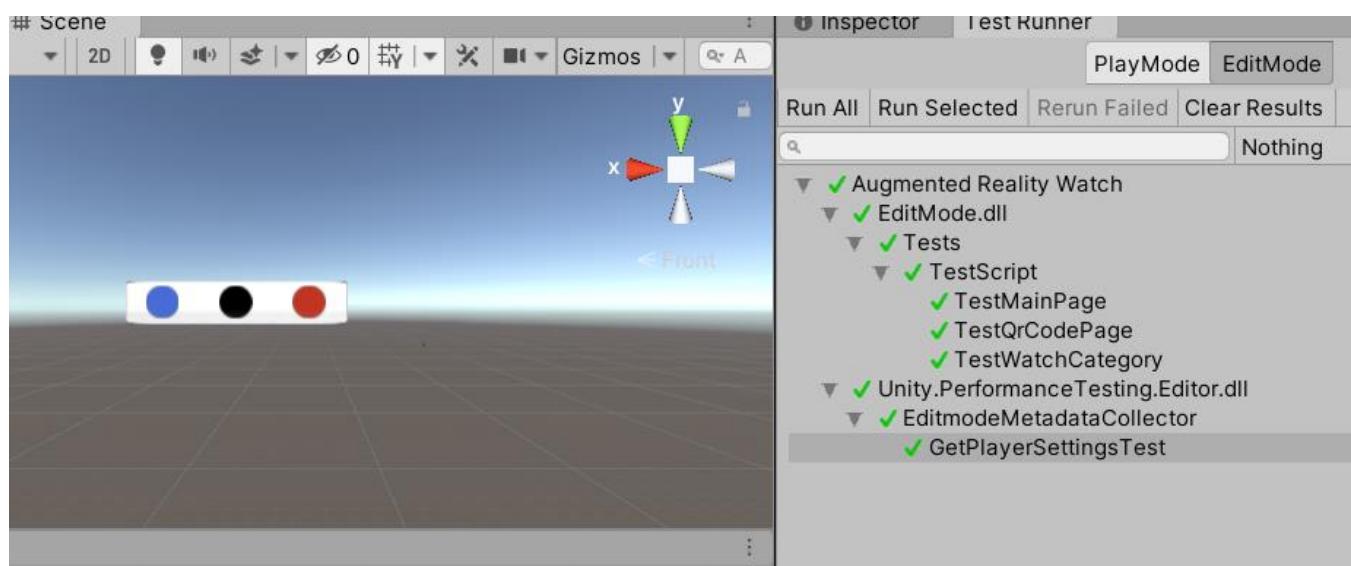
```
[Test]
public void TestQrCodePage()
{
    // Use the Assert class to test conditions
    var go = new GameObject("WatchesPages");
    Assert.AreEqual("QR_page", go.start("QR_page"));
}
```

CODE IS DEVELOPED

```
case "QR_page":
    page1.gameObject.SetActive(false);
    page2.gameObject.SetActive(false);
    W_page1.gameObject.SetActive(false);
    W_page2.gameObject.SetActive(false);
    W_page3.gameObject.SetActive(false);
    W_page4.gameObject.SetActive(false);
    QR_page.gameObject.SetActive(true);

    break;
```

CODE IS PASSED



7.2.5. TEST CASE 5 – FEATURE: LEATHER WATCH

TC005	Leather Watch	TC004	Click on “Leather Watch”	User will be directed to the leather watch gallery page	Passed
-------	---------------	-------	--------------------------	---	--------

TESTING CODE

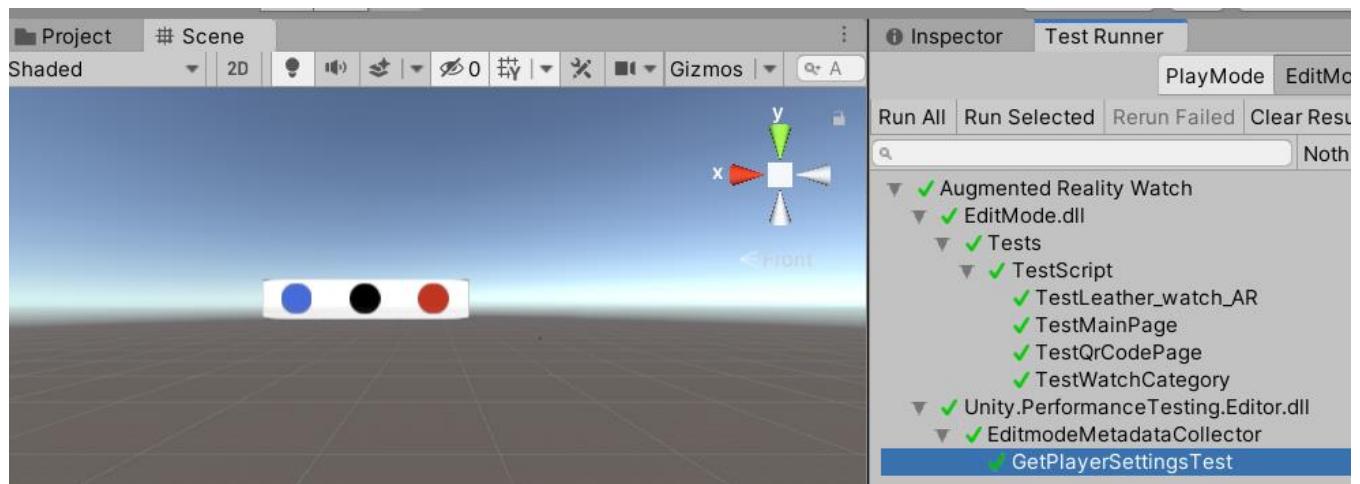
```
[Test]
public void TestLeather_watch_AR()
{
    // Use the Assert class to test conditions
    var go = new GameObject("WatchesPages");
    Assert.AreEqual("Leather_watch_AR", go.start("Leather_watch_AR"));
}
```

CODE IS DEVELOPED

```
case "Leather_watch_AR":
    page1.gameObject.SetActive(false);
    page2.gameObject.SetActive(false);
    W_page1.gameObject.SetActive(true);
    W_page2.gameObject.SetActive(false);
    W_page3.gameObject.SetActive(false);
    W_page4.gameObject.SetActive(false);
    QR_page.gameObject.SetActive(false);

    break;
```

CODE IS PASSED



7.2.6. TEST CASE 6 – FEATURE: SMART WATCH

TC006	Smart Watch	TC004	Click on “Smart Watch”	User will be directed to the leather watch gallery page	Passed
-------	-------------	-------	------------------------	---	--------

TESTING CODE

```
[Test]
public void TestSmart_watch_AR()
{
    // Use the Assert class to test conditions
    var go = new GameObject("WatchesPages");
    Assert.AreEqual("Smart_watch_AR", go.start("Smart_watch_AR"));
}
```

CODE IS DEVELOPED

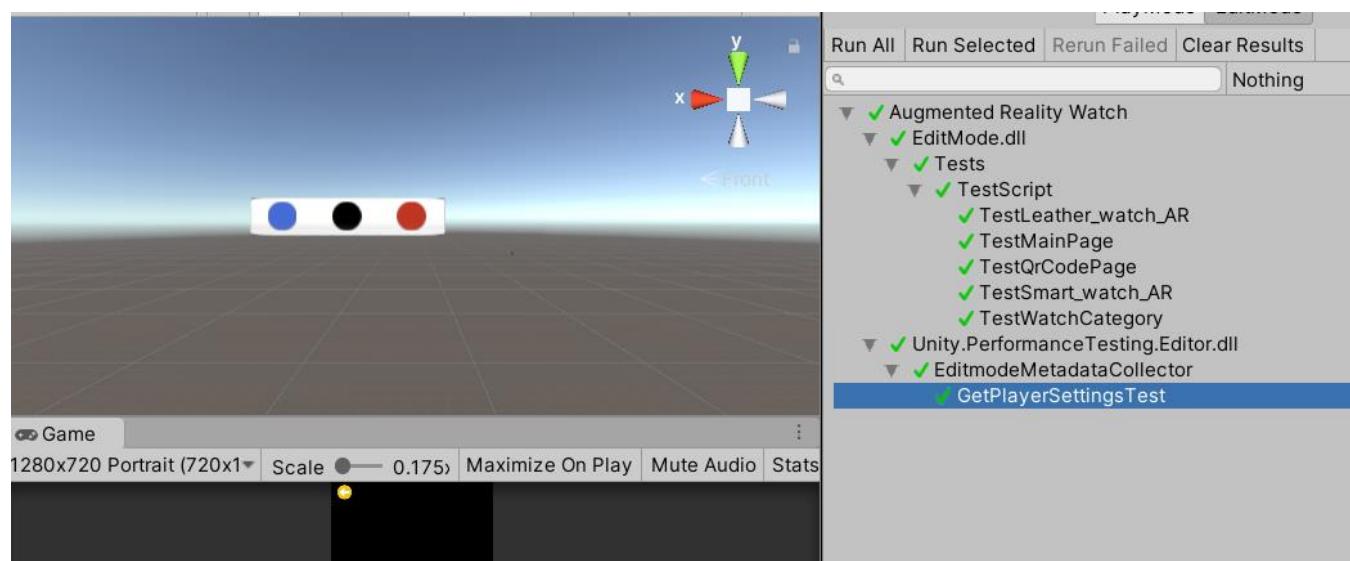
```

case "Smart_watch_AR":
    page1.gameObject.SetActive(false);
    page2.gameObject.SetActive(false);
    W_page1.gameObject.SetActive(false);
    W_page2.gameObject.SetActive(true);
    W_page3.gameObject.SetActive(false);
    W_page4.gameObject.SetActive(false);
    QR_page.gameObject.SetActive(false);

    break;

```

CODE IS PASSED



7.2.7. TEST CASE 7 – FEATURE: METAL WATCH

TC007	Metal Watch	TC004	Click on “Metal Watch”	User will be directed to the Metal watch gallery page	Passed
-------	-------------	-------	------------------------	---	--------

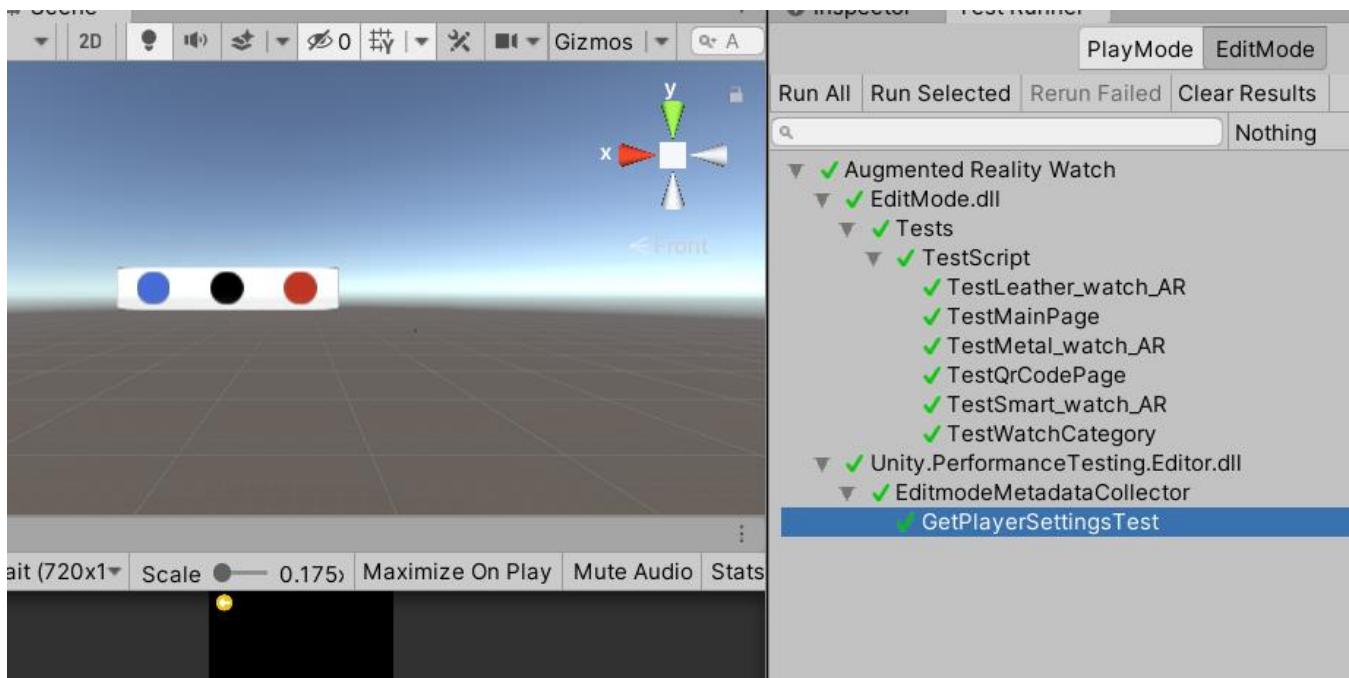
TESTING CODE

```
[Test]
public void TestMetal_watch_AR()
{
    // Use the Assert class to test conditions
    var go = new GameObject("WatchesPages");
    Assert.AreEqual("Metal_watch_AR", go.start("Metal_watch_AR"));
}
```

CODE IS DEVELOPED

```
case "Metal_watch_AR":
    page1.gameObject.SetActive(false);
    page2.gameObject.SetActive(false);
    W_page1.gameObject.SetActive(false);
    W_page2.gameObject.SetActive(false);
    W_page3.gameObject.SetActive(true);
    W_page4.gameObject.SetActive(false);
    QR_page.gameObject.SetActive(false);
```

CODE IS PASSED



7.2.8. TEST CASE 8 – FEATURE: SPORTS WATCH

TC008	Sports Watch	TC004	Click on “Sports Watch”	User will be directed to the Sports watch gallery page	Passed
-------	--------------	-------	-------------------------	--	--------

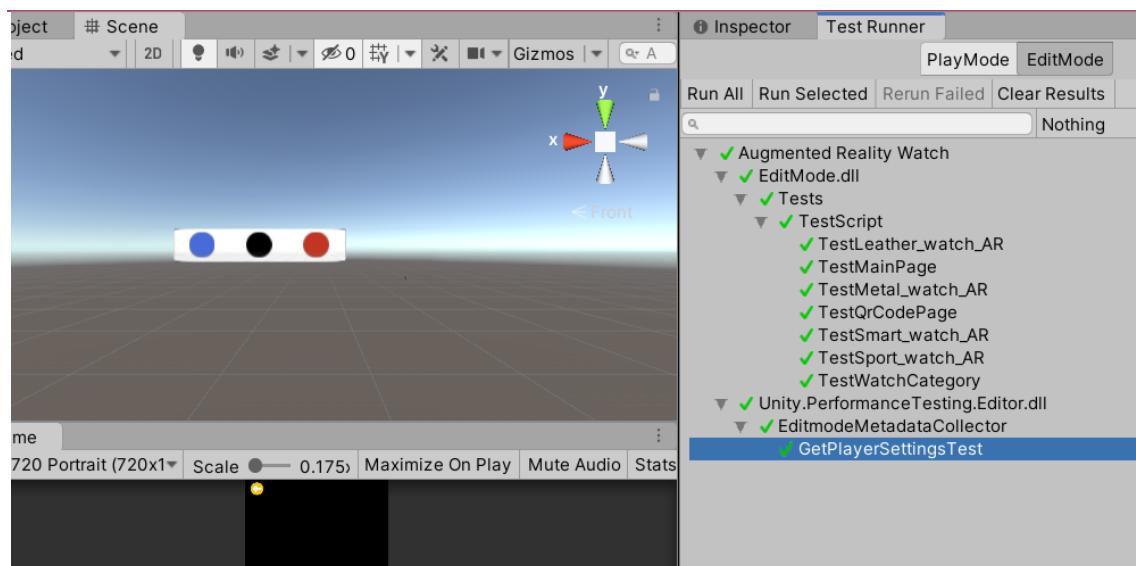
TESTING CODE

```
[Test]
public void TestSport_watch_AR()
{
    // Use the Assert class to test conditions
    var go = new GameObject("WatchesPages");
    Assert.AreEqual("Sport_watch_AR", go.start("Sport_watch_AR"));
}
```

CODE IS DEVELOPED

```
case "Sport_watch_AR":  
    page1.gameObject.SetActive(false);  
    page2.gameObject.SetActive(false);  
    W_page1.gameObject.SetActive(false);  
    W_page2.gameObject.SetActive(false);  
    W_page3.gameObject.SetActive(false);  
    W_page4.gameObject.SetActive(true);  
    QR_page.gameObject.SetActive(false);  
  
    break;
```

CODE IS PASSED



7.2.9. TEST CASE 9 – FEATURE: WATCH SELECTION

TC009	Watch Selection	TC005/TC00/	Select the watch	Camera will open with the back button on the top and colour	Passed
-------	-----------------	-------------	------------------	---	--------

		TC007/TC008	available in the gallery	selection on the middle right side of the camera	
--	--	-------------	--------------------------	--	---

TESTING CODE

```
[Test]
public void TestWatch_Selection()
{
    // Use the Assert class to test conditions
    WatchSelect go1 = new WatchSelect();
    // Use the Assert class to test conditions
    WatchSelect go2 = new WatchSelect();
    // Use the Assert class to test conditions
    WatchSelect go3 = new WatchSelect();
    // Use the Assert class to test conditions
    WatchSelect go4 = new WatchSelect();

    go1.WatchButtonOneClicked();
    go2.WatchButtonTwoClicked();
    go3.WatchButtonThreeClicked();
    go4.WatchButtonFourClicked();

    Assert.AreEqual(go1.wWindow1Animation.Play() == "1" && go2.wWindow2Animation.Play() == "1"
        && go3.wWindow3Animation.Play() == "1" && go4.wWindow4Animation.Play() == "1");
}
```

CODE IS DEVELOPED

```
public void WatchButtonOneClicked()
{
    //showing model of watch one on the wrist of user
    wModel1.SetActive(true);
    wModel2.SetActive(false);
    wModel3.SetActive(false);
    wModel4.SetActive(false);

    //animating the window of watch one
    wWindow1Animation["w1animation"].speed = 1;
    wWindow1Animation.Play();
}

public void WatchButtonTwoClicked()
{
    //showing model of watch two on the wrist of user
    wModel1.SetActive(false);
    wModel2.SetActive(true);
    wModel3.SetActive(false);
    wModel4.SetActive(false);

    //animating the window of watch two
    wWindow2Animation["w2animation"].speed = 1;
    wWindow2Animation.Play();
}

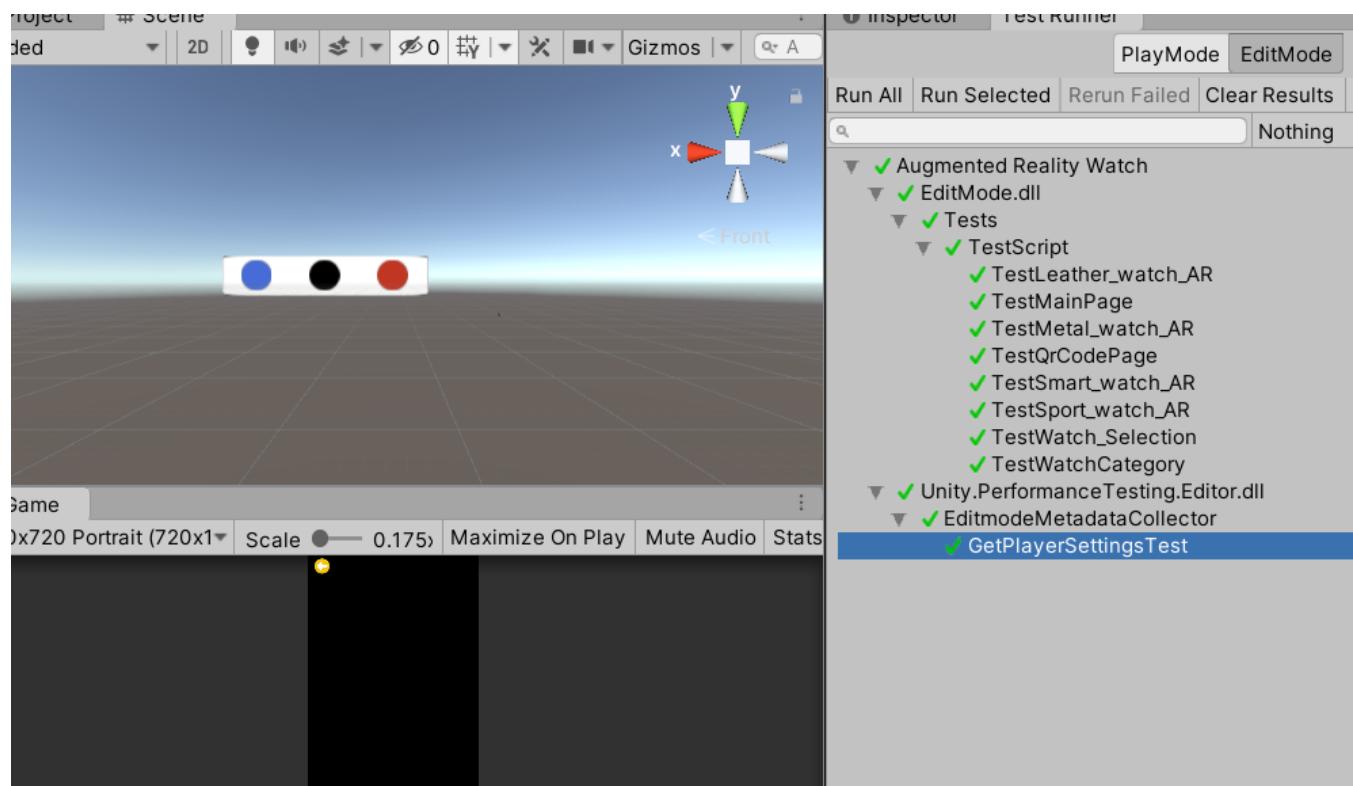
public void WatchButtonThreeClicked()
{
    //showing model of watch three on the wrist of user
    wModel1.SetActive(false);
    wModel2.SetActive(false);
    wModel3.SetActive(true);
    wModel4.SetActive(false);

    //animating the window of watch three
    wWindow3Animation["w3animation"].speed = 1;
    wWindow3Animation.Play();
}

public void WatchButtonFourClicked()
{
    //showing model of watch four on the wrist of user
    wModel1.SetActive(false);
    wModel2.SetActive(false);
    wModel3.SetActive(false);
    wModel4.SetActive(true);

    //animating the window of watch four
}
```

CODE IS PASSED



7.2.10. TEST CASE 10 – FEATURE: AR VIEW

TC010	AR View	TC009	Scan the marker to view the 3D model of the watch	The 3D model of the watch will be displayed	Passed
-------	---------	-------	---	---	--------

TESTING CODE

This test case is not automated because it is not possible to check in unity with automated test code. Therefore, this part tested manually.

CODE IS DEVELOPED

```
public void WatchButtonOneClicked()
{
    //showing model of watch one on the wrist of user
    wModel1.SetActive(true);
    wModel2.SetActive(false);
    wModel3.SetActive(false);
    wModel4.SetActive(false);

    //animating the window of watch one
    wWindow1Animation["w1animation"].speed = 1;
    wWindow1Animation.Play();
}

public void WatchButtonTwoClicked()
{
    //showing model of watch two on the wrist of user
    wModel1.SetActive(false);
    wModel2.SetActive(true);
    wModel3.SetActive(false);
    wModel4.SetActive(false);

    //animating the window of watch two
    wWindow2Animation["w2animation"].speed = 1;
    wWindow2Animation.Play();
}

public void WatchButtonThreeClicked()
{
    //showing model of watch three on the wrist of user
    wModel1.SetActive(false);
    wModel2.SetActive(false);
    wModel3.SetActive(true);
    wModel4.SetActive(false);

    //animating the window of watch three
    wWindow3Animation["w3animation"].speed = 1;
    wWindow3Animation.Play();
}

public void WatchButtonFourClicked()
{
    //showing model of watch four on the wrist of user
    wModel1.SetActive(false);
    wModel2.SetActive(false);
    wModel3.SetActive(false);
    wModel4.SetActive(true);

    //animating the window of watch four
}
```

CODE IS PASSED



7.2.11. TEST CASE 11 – FEATURE: CHANGING THE WATCH

TC011	Changing the watch	TC010	Click back button on the left top corner	Watch gallery page will be displayed and user can select the watch or go back and change the category view a different type of watch	Passed
-------	--------------------	-------	--	--	--------

TESTING CODE

This test case is not automated because it is not possible to check in unity with automated test code. Therefore, this part tested manually.

CODE IS DEVELOPED

```
public void WatchButtonOneClicked()
{
    //showing model of watch one on the wrist of user
    wModel1.SetActive(true);
    wModel2.SetActive(false);
    wModel3.SetActive(false);
    wModel4.SetActive(false);

    //animating the window of watch one
    wWindow1Animation["w1animation"].speed = 1;
    wWindow1Animation.Play();
}

public void WatchButtonTwoClicked()
{
    //showing model of watch two on the wrist of user
    wModel1.SetActive(false);
    wModel2.SetActive(true);
    wModel3.SetActive(false);
    wModel4.SetActive(false);

    //animating the window of watch two
    wWindow2Animation["w2animation"].speed = 1;
    wWindow2Animation.Play();
}

public void WatchButtonThreeClicked()
{
    //showing model of watch three on the wrist of user
    wModel1.SetActive(false);
    wModel2.SetActive(false);
    wModel3.SetActive(true);
    wModel4.SetActive(false);

    //animating the window of watch three
    wWindow3Animation["w3animation"].speed = 1;
    wWindow3Animation.Play();
}

public void WatchButtonFourClicked()
{
    //showing model of watch four on the wrist of user
    wModel1.SetActive(false);
    wModel2.SetActive(false);
    wModel3.SetActive(false);
    wModel4.SetActive(true);

    //animating the window of watch four
}
```

CODE IS PASSED



7.2.12. TEST CASE 12 – FEATURE: CHANGING THE COLOUR OF WATCH

TC012	Changing the colour of watch	TC010	Choose the colour available on the AR view	The watch colour will be changed accordingly if available for that model	Passed
-------	------------------------------	-------	--	--	--------

TESTING CODE

```
[Test]
public void TestColourSelect()
{
    // Use the Assert class to test conditions
    ColourSelect go1 = new ColourSelect();
    // Use the Assert class to test conditions
    ColourSelect go2 = new ColourSelect();
    // Use the Assert class to test conditions
    ColourSelect go3 = new ColourSelect();
    // Use the Assert class to test conditions
    ColourSelect go4 = new ColourSelect();

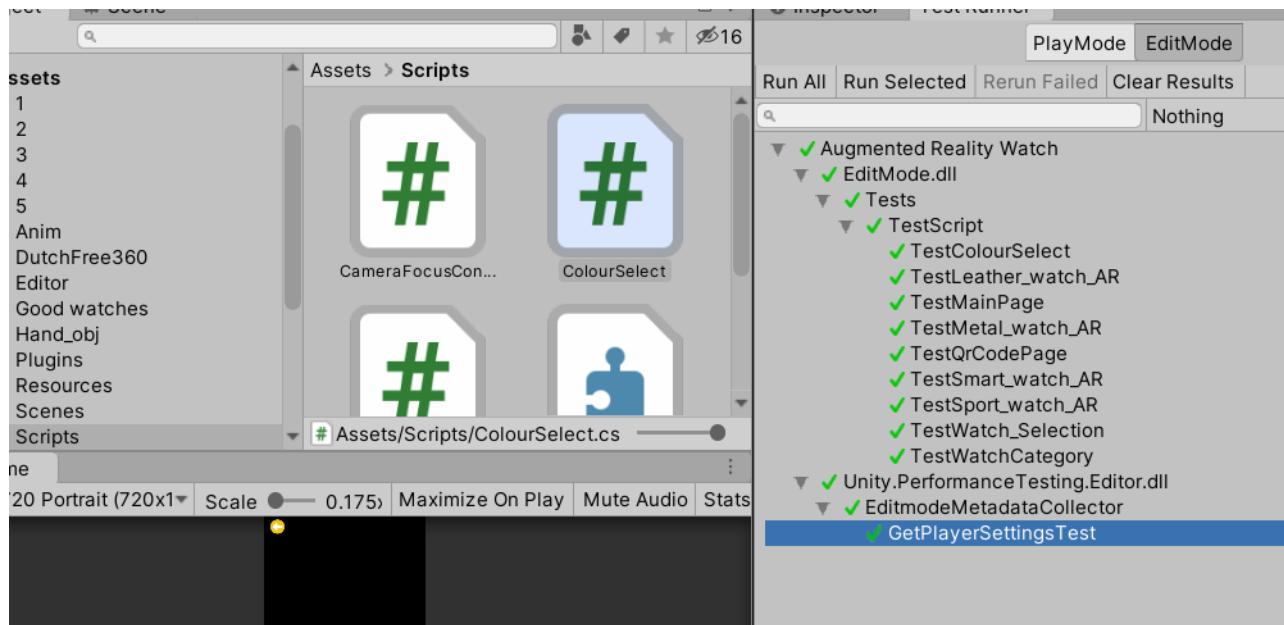
    go1.watchColourSelect().nameOfButton = "Blue button";
    go2.watchColourSelect().nameOfButton = "White button";
    go3.watchColourSelect().nameOfButton = "Black button";
    go4.watchColourSelect().nameOfButton = "Red button";

    Assert.AreEqual(go1.watchColourSelect().myColor == "#103B90" && go2.watchColourSelect().myColor == "#A1A1A1"
        && go3.watchColourSelect().myColor == "#000000" && go4.watchColourSelect().myColor == "#740202");
}
```

CODE IS DEVELOPED

```
public void watchColourSelect()
{
    string nameOfButton = EventSystem.current.currentSelectedGameObject.name;
    if(nameOfButton == "Blue button")
    {
        //colour of strap will change to blue
        Color myColor = new Color();
        ColorUtility.TryParseHtmlString("#103B90", out myColor);
        band.color = myColor;
    }
    else if (nameOfButton == "White button")
    {
        //colour of strap will change to White
        Color myColor = new Color();
        ColorUtility.TryParseHtmlString("#A1A1A1", out myColor);
        band.color = myColor;
    }
    else if(nameOfButton == "Black button")
    {
        //colour of strap will change to black
        Color myColor = new Color();
        ColorUtility.TryParseHtmlString("#000000", out myColor);
        band.color = myColor;
    }
    else if (nameOfButton == "Red button")
    {
        //colour of strap will change to red
        Color myColor = new Color();
        ColorUtility.TryParseHtmlString("#740202", out myColor);
        band.color = myColor;
    }
}
```

CODE IS PASSED



7.2.13. OTHER TESTING SAMPLES

Here we present you the sample screenshots of the test case sample which we have performed on this application

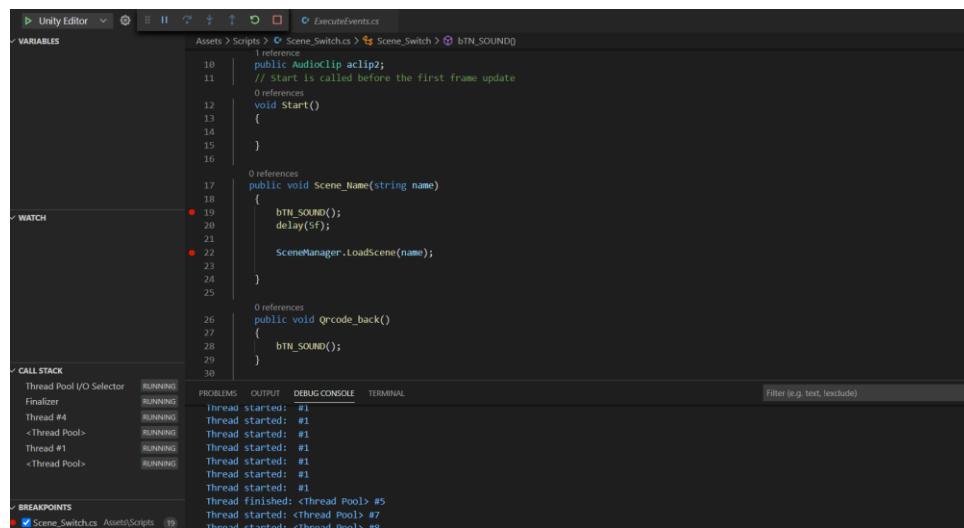


Figure 31

After the application is opened and when a click action is performed inside in the app the event system will recognise which part has been clicked in the application and will navigate to the respective page accordingly.

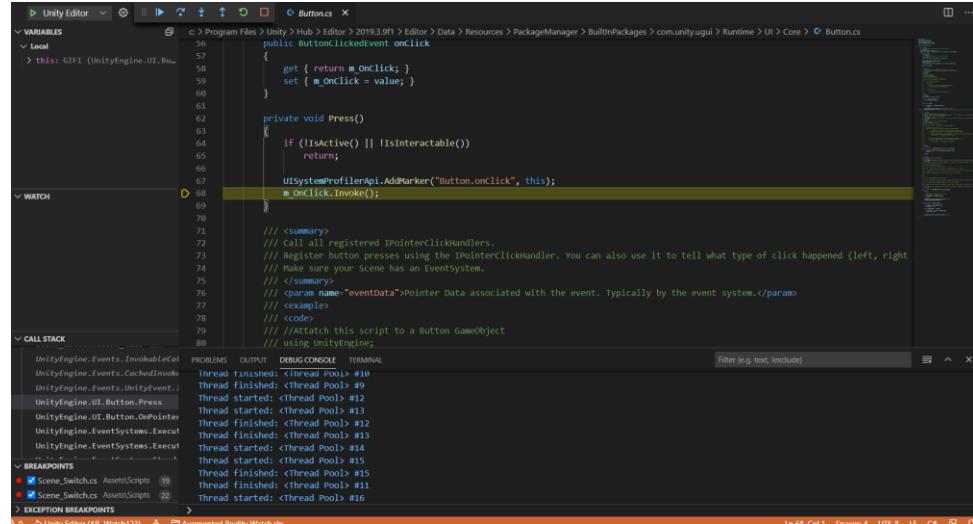


Figure 32

Here we can check whether the correct variables are called based on the respective event which is highlighted in below figure.

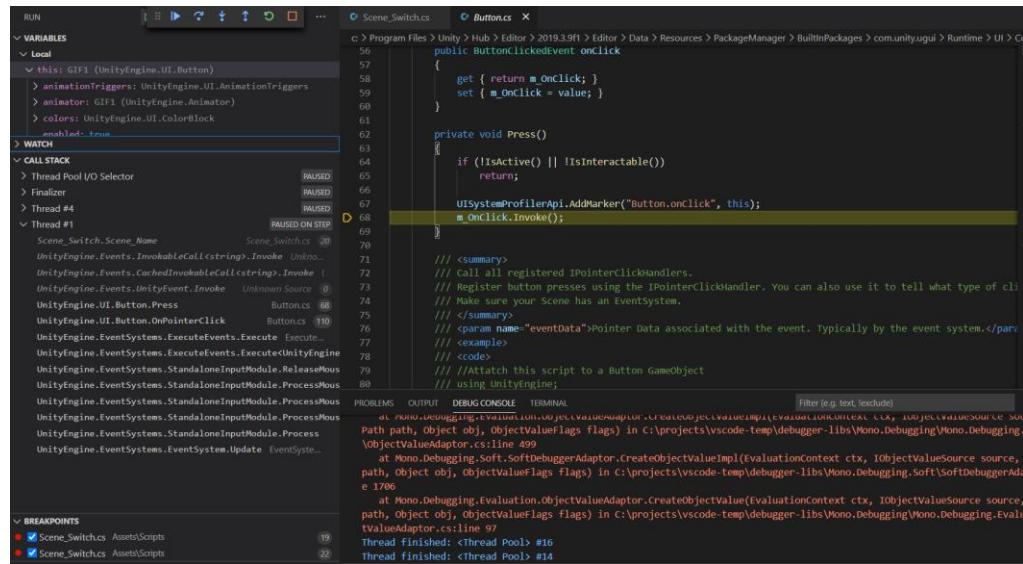


Figure 33

Below figure shows the functionality and work flow of application and each components response has been tested manually during runtime.

The screenshot shows the Visual Studio IDE interface for a Unity project named "AR WATCH123". The code editor displays C# script "Manager2.cs" with several breakpoints marked by red dots. The script contains methods for sound playback and scene loading based on button selection. The output window at the bottom shows the process of installing ".NET Core Debugger" and "Razor Language Server" packages.

```

public void bTN_SOUND()
{
    source.Stop();
    source.PlayOneShot(aclip2);
}

public void select_leather_watch1(int num)
{
    if (num == 0)
    {
        bTN_SOUND();
        PlayerPrefs.SetInt("Leather", num);
        SceneManager.LoadScene("Leather_watch_AR");
    }
    if (num == 1)
    {
        bTN_SOUND();
        PlayerPrefs.SetInt("Leather", num);
        SceneManager.LoadScene("Leather_watch_AR");
    }
}

```

Figure 34

We have performed Category Selection where the sports watch has been selected and its response has been verified in the runtime which is shown in above figure.

The screenshot shows the Visual Studio Code interface with the title "DefaultTrackableEventHandler.cs - AR Watch123 - Visual Studio Code [Administrator]". The code editor displays the "DefaultTrackableEventHandler.cs" file from the Unity project. The left sidebar shows the "WATCH" and "CALL STACK" panes, indicating the application is running. The status bar at the bottom shows assembly loading information and the current file path.

```

protected virtual void OnTrackingFound()
{
    if (!mTrackableBehaviour)
    {
        var renderComponents = mTrackableBehaviour.GetComponentInChildren(true);
        var colliderComponents = mTrackableBehaviour.GetComponentInChildren(true);
        var canvasComponents = mTrackableBehaviour.GetComponentInChildren(true);

        // Double rendering:
        foreach (var component in renderComponents)
            component.enabled = true;

        // Enable colliders:
        foreach (var component in colliderComponents)
            component.enabled = true;

        // Enable canvas:
        foreach (var component in canvasComponents)
            component.enabled = true;
    }
}

```

Figure 35

Above figure shows the activity where the user tries to scan the AR marker during runtime. Also this testing has been performed on two conditions. First condition where the scanned image target is matching with the image uploaded in the vuforia cloud. Second condition is where the scanned image target is not matching with image target present in the vuforia cloud.



Figure 36

Above figure shows the error message when vuforia camera was not initiated when the user tries to scan the AR marker during runtime

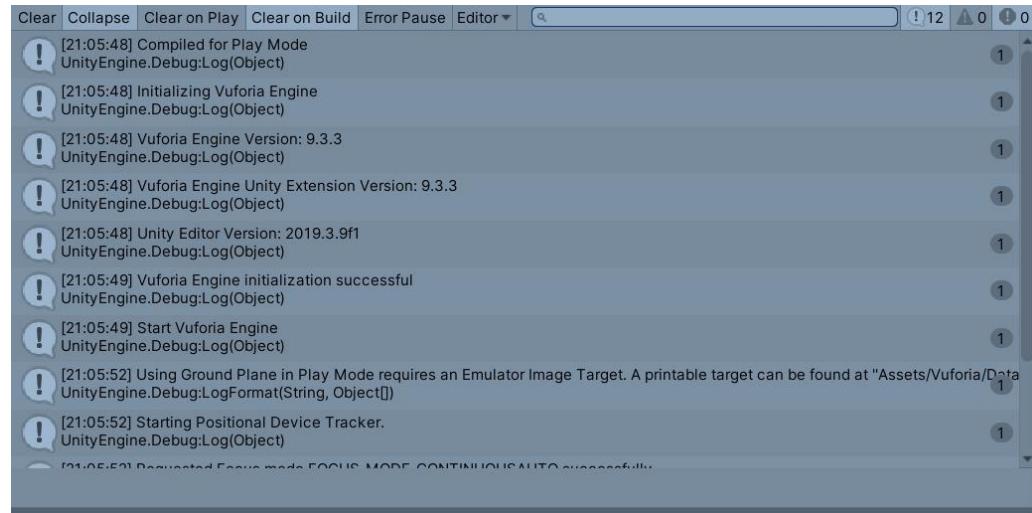


Figure 37

This figure shows the runtime message when vuforia camera was initiated when the user tries to scan the AR marker

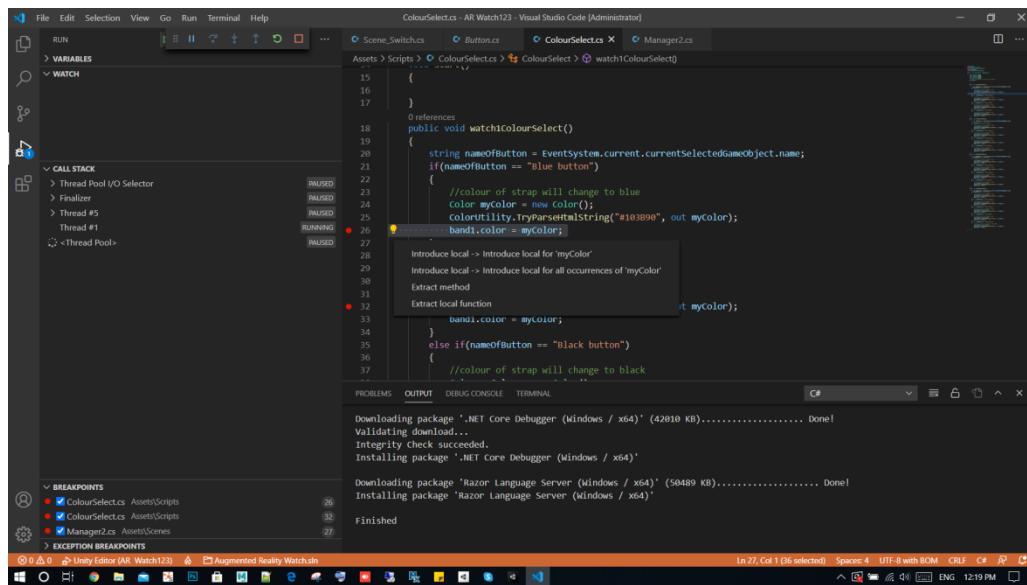


Figure 38

Above figure represents the colour functionality where the user is able to change the colour of the watch selected in augmented space during runtime.

7.3. CODE REFACTORING

In this project, most of the development was creating game object visually. Therefore, there was no any big coding parts that can be refactored.

7.3.1. CODE SMELL 1: DUPLICATED CODE

BEFORE

```
public void WatchButtonOneClicked()
{
    //showing model of watch one on the wrist of user
    wModel1.SetActive(true);
    wModel2.SetActive(false);
    wModel3.SetActive(false);
    wModel4.SetActive(false);

    //animating the window of watch one
    wWindow1Animation["w1animation"].speed = 1;
    wWindow1Animation.Play();
}

public void WatchButtonTwoClicked()
{
    //showing model of watch two on the wrist of user
    wModel1.SetActive(false);
    wModel2.SetActive(true);
    wModel3.SetActive(false);
    wModel4.SetActive(false);

    //animating the window of watch two
    wWindow2Animation["w2animation"].speed = 1;
    wWindow2Animation.Play();
}

public void WatchButtonThreeClicked()
{
    //showing model of watch three on the wrist of user
    wModel1.SetActive(false);
    wModel2.SetActive(false);
    wModel3.SetActive(true);
    wModel4.SetActive(false);

    //animating the window of watch three
    wWindow3Animation["w3animation"].speed = 1;
    wWindow3Animation.Play();
}

public void WatchButtonFourClicked()
{
    //showing model of watch four on the wrist of user
    wModel1.SetActive(false);
    wModel2.SetActive(false);
    wModel3.SetActive(false);
    wModel4.SetActive(true);
```

AFTER

```
public void setModel(string one, string two, string three, string four
{
    //showing model of watch one on the wrist of user
    wModel1.SetActive(one);
    wModel2.SetActive(two);
    wModel3.SetActive(three);
    wModel4.SetActive(four);
}
public void WatchButtonOneClicked()
{
    setModel(true, false, false, false);
    //animating the window of watch one
    wWindow1Animation["w1animation"].speed = 1;
    wWindow1Animation.Play();
}
public void WatchButtonTwoClicked()
{
    setModel(false, true, false, false);
    //animating the window of watch two
    wWindow2Animation["w2animation"].speed = 1;
    wWindow2Animation.Play();
}
public void WatchButtonThreeClicked()
{
    setModel(false, false, true, false);
    //animating the window of watch three
    wWindow3Animation["w3animation"].speed = 1;
    wWindow3Animation.Play();
}
public void WatchButtonFourClicked()
{
    setModel(false, false, false, true);
    //animating the window of watch four
    wWindow4Animation["w4animation"].speed = 1;
    wWindow4Animation.Play();
}
```

7.3.2. CODE SMELL 2: REMOVE DUBLICATION

BEFORE

```
public void select_Leather_watch1(int num)
{
    if (num == 0)
    {
        bTN_SOUND();
        PlayerPrefs.SetInt("Leather", num);
        SceneManager.LoadScene("Leather_watch_AR");
    }

    if (num == 1)
    {
        bTN_SOUND();
        PlayerPrefs.SetInt("Leather", num);
        SceneManager.LoadScene("Leather_watch_AR");
    }

    if (num == 2)
    {
        bTN_SOUND();
        PlayerPrefs.SetInt("Leather", num);
        SceneManager.LoadScene("Leather_watch_AR");
    }

    if (num == 3)
    {
        bTN_SOUND();
        PlayerPrefs.SetInt("Leather", num);
        SceneManager.LoadScene("Leather_watch_AR");
    }

    // Update is called once per frame
}
}
```

AFTER

```
public void select_Leather_watch1(int num)
{
    bTN_SOUND();
    PlayerPrefs.SetInt("Leather", num);
    SceneManager.LoadScene("Leather_watch_AR");
}
```

7.3.3. CODE SMELL 3: UNCLEAR FUNCTION NAME / RENAME

BEFORE

```
public void Active(string name)
{
    if (name == "Leather_watch_AR")
    {
        PlayerPrefs.SetString("namepage", name);
    }
    if (name == "Metal_watch_AR")
    {
        PlayerPrefs.SetString("namepage", name);
    }
    if (name == "Smart_watch_AR")
    {
        PlayerPrefs.SetString("namepage", name);
    }
    if (name == "Sport_watch_AR")
    {
        PlayerPrefs.SetString("namepage", name);
    }
}
```

AFTER

```

public void ActivePages(string Active_Page_Name)
{
    if (Active_Page_Name == "Leather_watch_AR")
    {
        PlayerPrefs.SetString("namepage", name);
    }
    if (Active_Page_Name == "Metal_watch_AR")
    {
        PlayerPrefs.SetString("namepage", name);
    }
    if (Active_Page_Name == "Smart_watch_AR")
    {
        PlayerPrefs.SetString("namepage", name);
    }
    if (Active_Page_Name == "Sport_watch_AR")
    {
        PlayerPrefs.SetString("namepage", name);
    }
}

```

7.4. TEST CASES FOR POST DEVELOPMENT

Test case ID	Feature	Preconditions	Test Procedure	Expected outputs
TC013	Android Adaptation	TC001	Install the application on android.	Application should be able to installed on the phone.
TC014	Start up	TC001, TC013	Click the application	Opening time should not

			and measure opening time.	exceed 5 seconds.
TC015	Battery time	TC001	Click the application and use application 5 minutes.	Application should not affect battery consumption should not exceed 3%.
TC016	Long term test	TC001	Open the application and wait 24 hours.	After 24 hours of application test run, there should not be any errors.
TC017	Memory consumption	TC001	Open the application and measure memory consumption.	Mobile application ram memory consumption should not be exceeded 500 MB.
TC018	Application size	TC001 - TC013	Measure application size.	Application size should not exceed 200 MB.
TC019	Hardware and Software variation	TC001 - TC013	Install application different phones.	Application should not have any problems.
TC020	Usage with other Applications	TC001 - TC013	Open application with other applications	While application is running it should not affect other application.
TC021	User Experience	TC001 - TC013	Black Box Testing	It should pass multiple black box testing with different

				people.
TC022	Security Testing	TC001 - TC013	Carry out security tests	

7.4.1. PERFORMANCE TESTING:

7.4.1.1. TEST CASE 13 – FEATURE: ANDROID COMPABILITY

TC013	Application must be able to run on Android	TC001	Install the application on android.	Application should be able to install on the phone.	Pass
-------	--	-------	-------------------------------------	---	------

This test has been manually done.

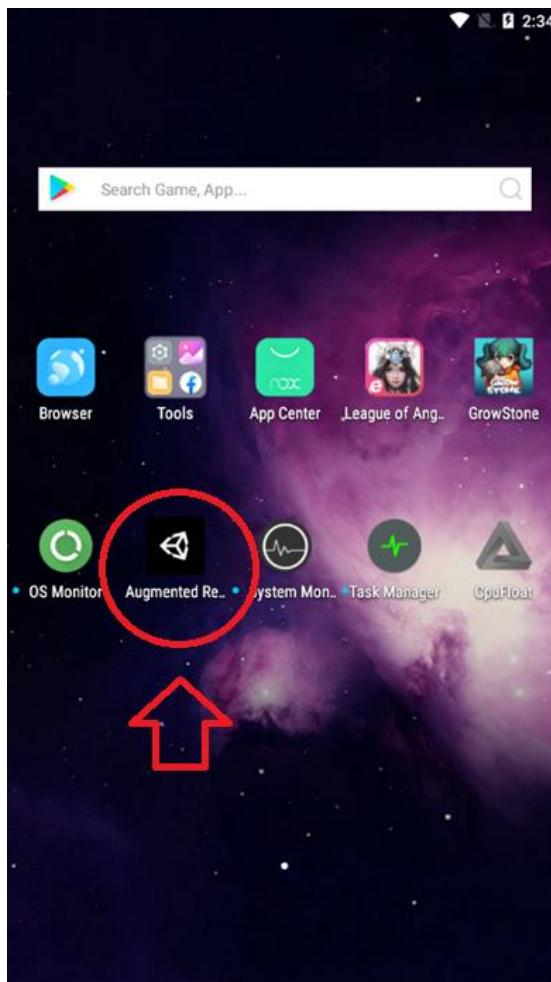


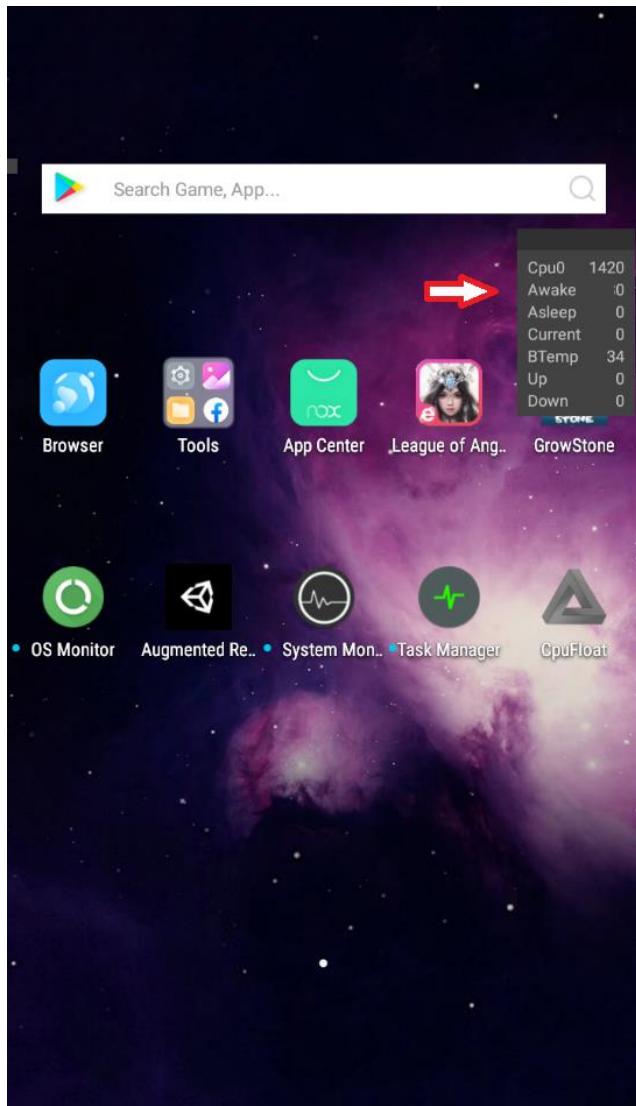
Figure 39: Application is Installed

7.4.1.2. TEST CASE 14 – FEATURE: START UP

BEFORE CODE REFACTORING

TC014	Start up	TC001, TC013	Click the application and measure opening time.	Opening time should not exceed 15 seconds.	Failed
-------	----------	-----------------	---	--	--------

In this test CpuFloat tool is used. It took 24 seconds to start up the AR Watch application. Therefore, it is failed from this test.



AR Watches

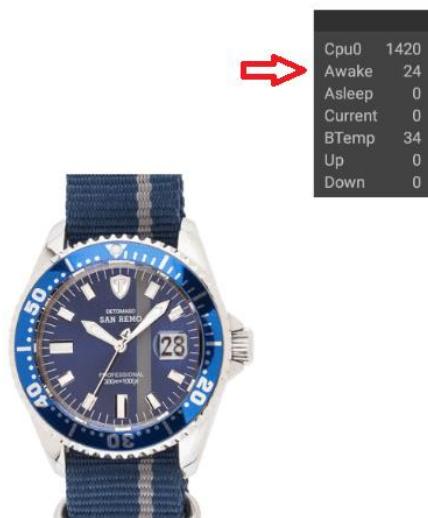
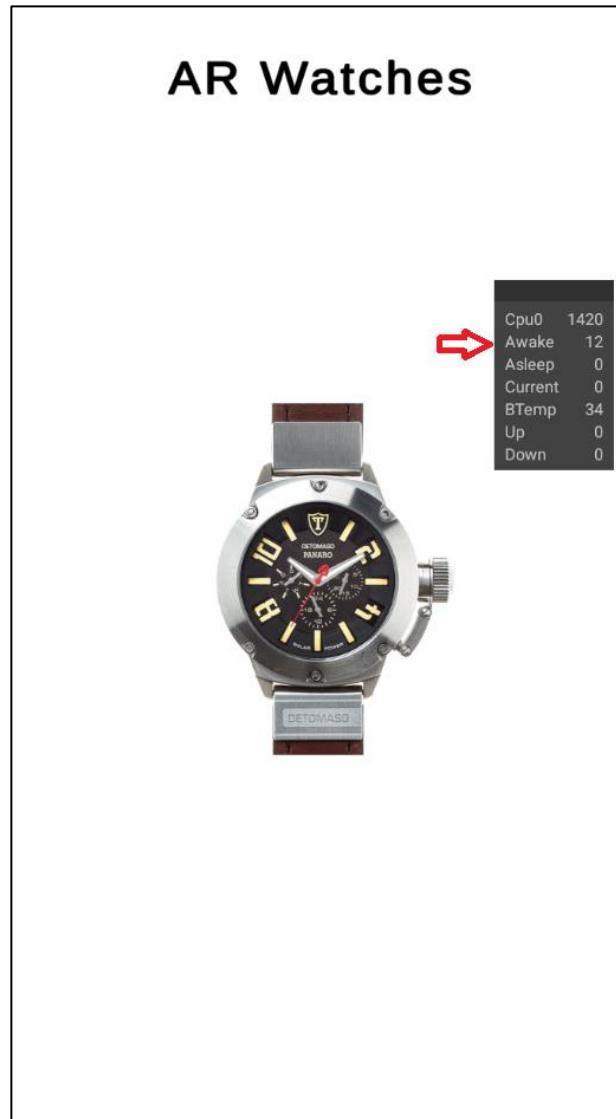


Figure 40: Application Start up

AFTER CODE REFACTORING

Number of images that shows in the beginning reduced to half.

TC014	Start up	TC001, TC013	Click the application and measure opening time.	Opening time should not exceed 15 seconds.	Pass
-------	----------	-----------------	---	--	------



7.4.1.3. TEST CASE 15 – FEATURE: BATTERY TIME

TC015	Battery time	TC001	Click the application and use application 5 minutes.	Application should not affect battery consumption should not exceed 3%.	Pass
-------	--------------	-------	--	---	------

After 5 minute of application usage only 1 percent battery level decreased. Phone battery analytics is used the measure performance.

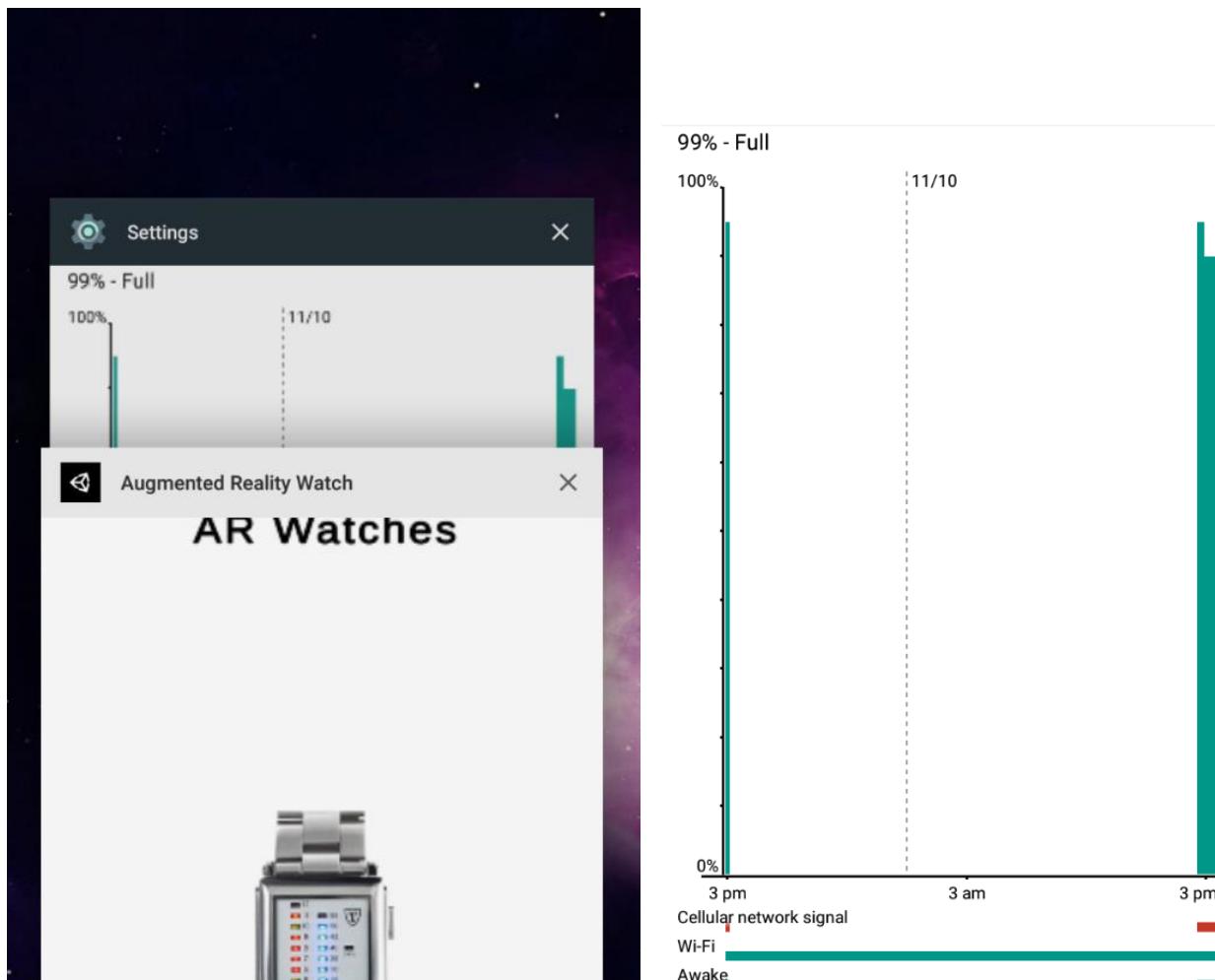


Figure 41: Battery Level

7.4.1.4. TEST CASE 16 – FEATURE: LONG TERM TEST

TC016	Long term test	TC001	Open the application and wait 24 hours.	After 24 hours of application test run, there should not be any errors.	Pass
-------	----------------	-------	---	---	------

No problem is accrued.

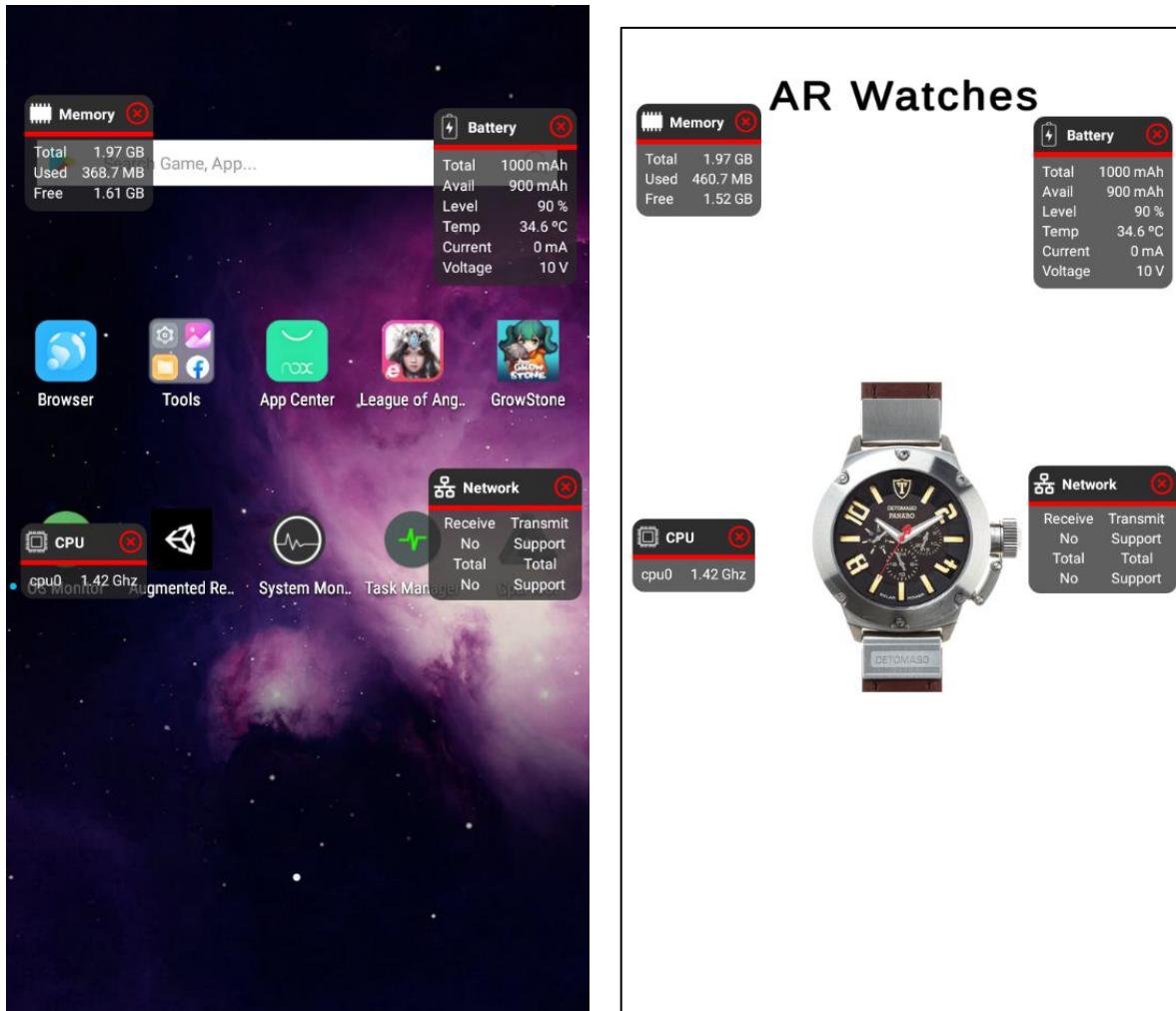


Figure 42: Long Term Run

7.4.1.5. TEST CASE 17 – FEATURE: MEMORY CONSUMPTION

TC017	Memory consumption	TC001	Open the application and measure memory consumption.	Mobile application ram memory consumption should not be exceeded 500 MB.	Pass
-------	--------------------	-------	--	--	------

In the beginning, used memory is 369 MB. After application started it 461 MB. Therefore, RAM usage is less than 100 MB. When the camera is on RAM usage increases 40 MB more but still it is less than 150 MB.



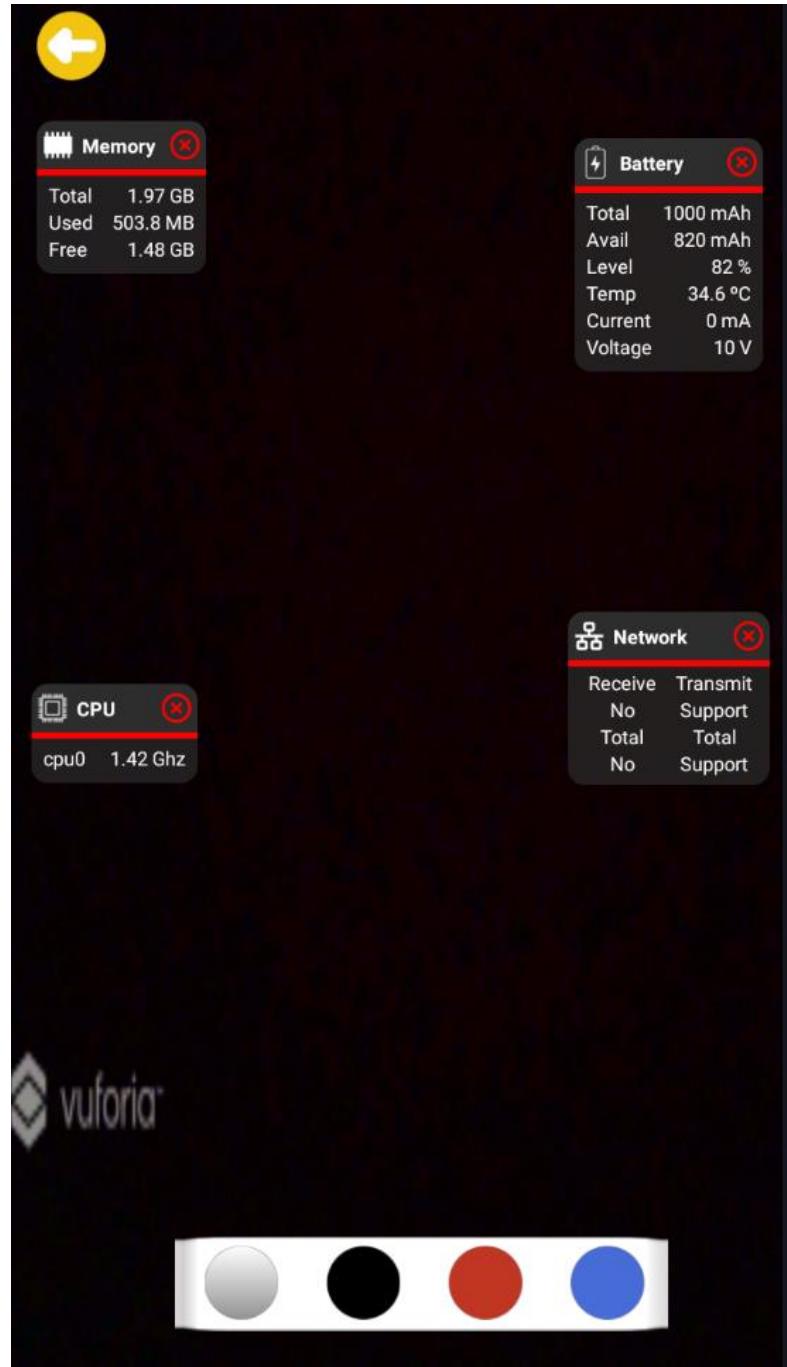


Figure 43: Ram Consumption Before Application and After Application is Opened

7.4.1.6. TEST CASE 18 – FEATURE: APPLICATION SIZE

TC018	Application size	TC001 - TC013	Measure application size.	Application size should not exceed 200 MB.	Pass
-------	------------------	---------------	---------------------------	--	------

Application size is 189 MB. It can be seen in build file in the application file. This test has to be check manually before application is released.

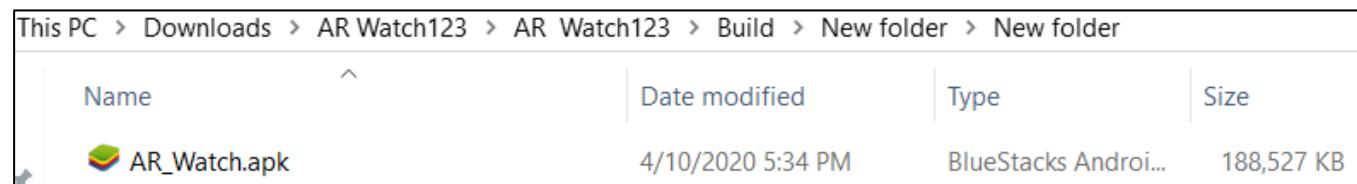


Figure 44: Application Size

7.4.1.7. TEST CASE 19 – FEATURE: HARDWARE AND SOFTWARE VARIATION

TC019	Hardware and Software variation	TC001 - TC013	Install application different phones.	Application should not have any problems.	Pass
-------	---------------------------------	---------------	---------------------------------------	---	------

All functional test cases are tested in 2 different android mobile phone. No problem is occurred. However, number of test for different phones should be increased to 10.

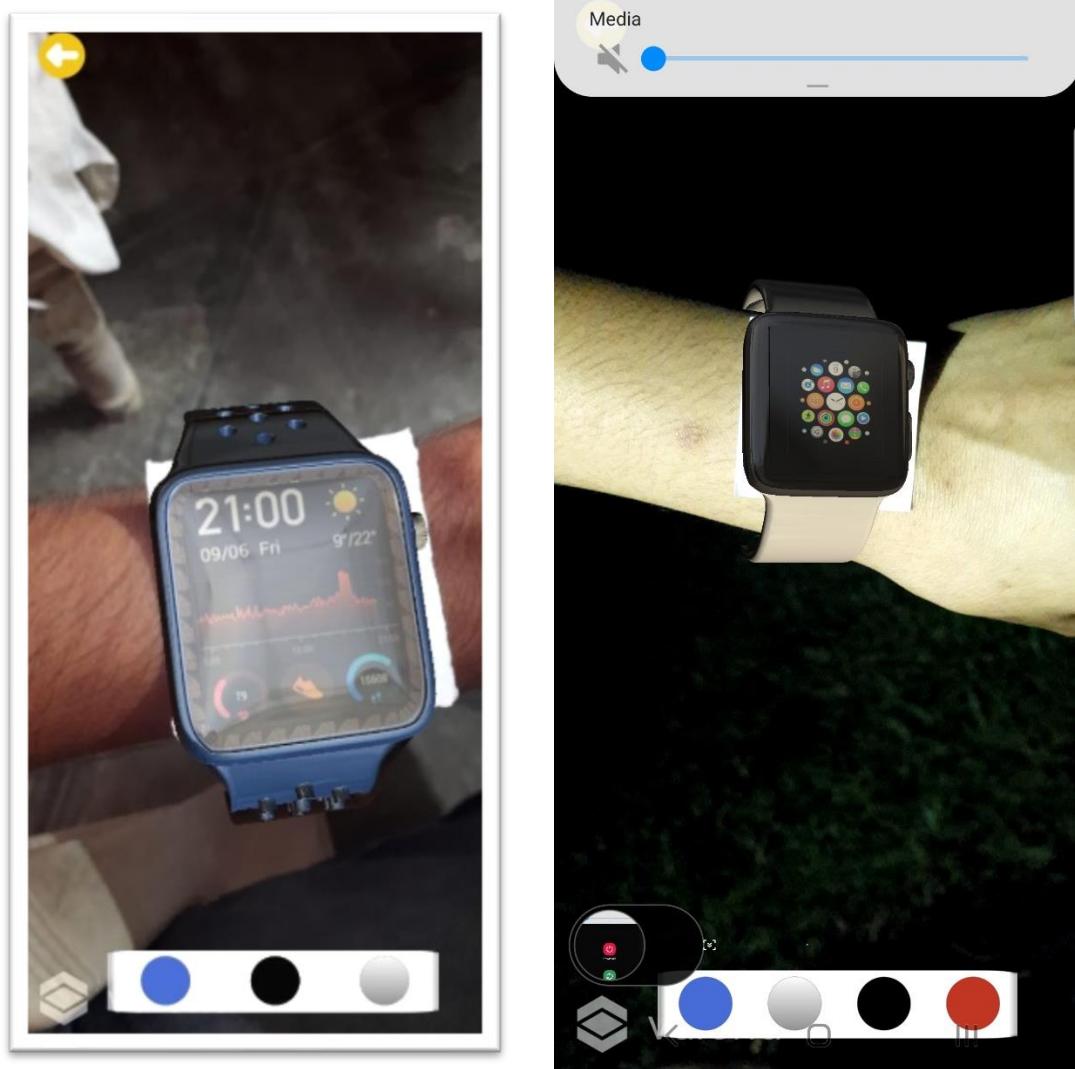


Figure 45: Different Mobile Phones

7.4.1.8. TEST CASE 20 – FEATURE: USAGE WITH OTHER APPLICATIONS

TC020	Usage with other Applications	TC001 - TC013	Open application with other applications	While application is running it should not affect other application.	pass
-------	-------------------------------	---------------	--	--	------

While application is background other applications are run and then after application is returned to front and no problem occurred.

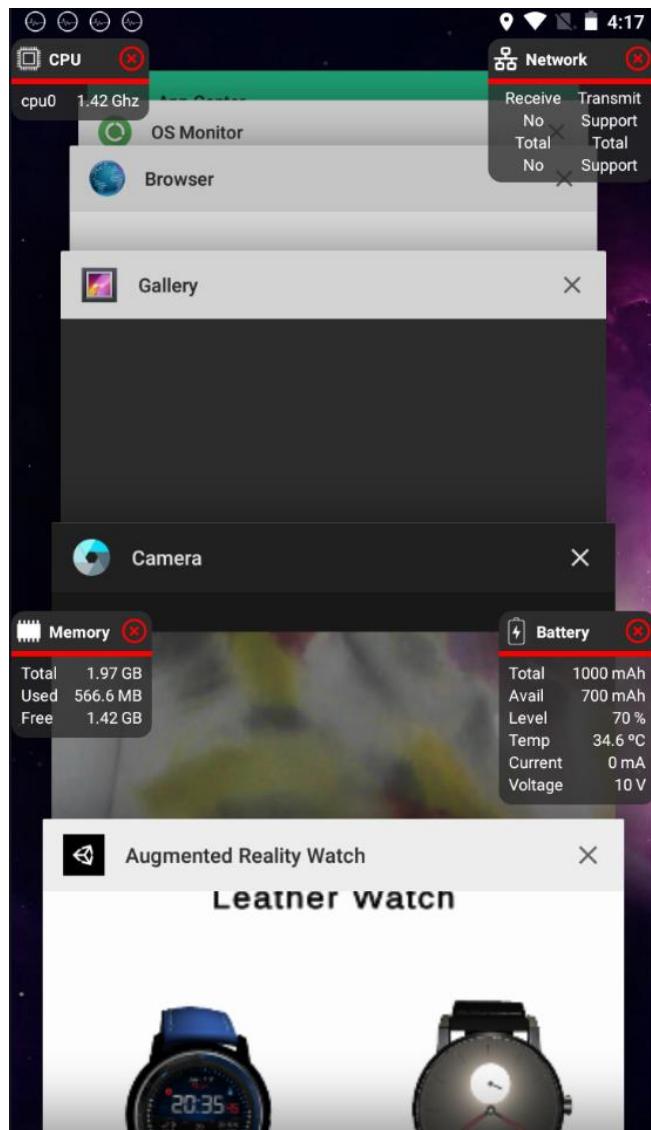


Figure 46: Testing with other application on background

7.4.2. SECURITY TESTING

In order to ensure that the application do not disclose the user information and is risk free we have followed the below mentioned security practices in this augmented reality application.

1. Enhanced data security
2. Key Management
3. Secure Communication
4. Cache Encryption
5. Code Obfuscation

7.4.2.1. ENHANCED DATA SECURITY

Data security policy and guidelines lay down by Android and iOS has been implemented in this application to ensure data leakage and also avoid getting under the radar of hackers.

7.4.2.2. KEY MANAGEMENT

Key management is very crucial for encryption because the hacker can easily gain control of the devices if hard coding keys are used for development. So, we have adopted the latest encryption standards and APIs, such as 256-bit encryption with SHA-256 hashing for this application.

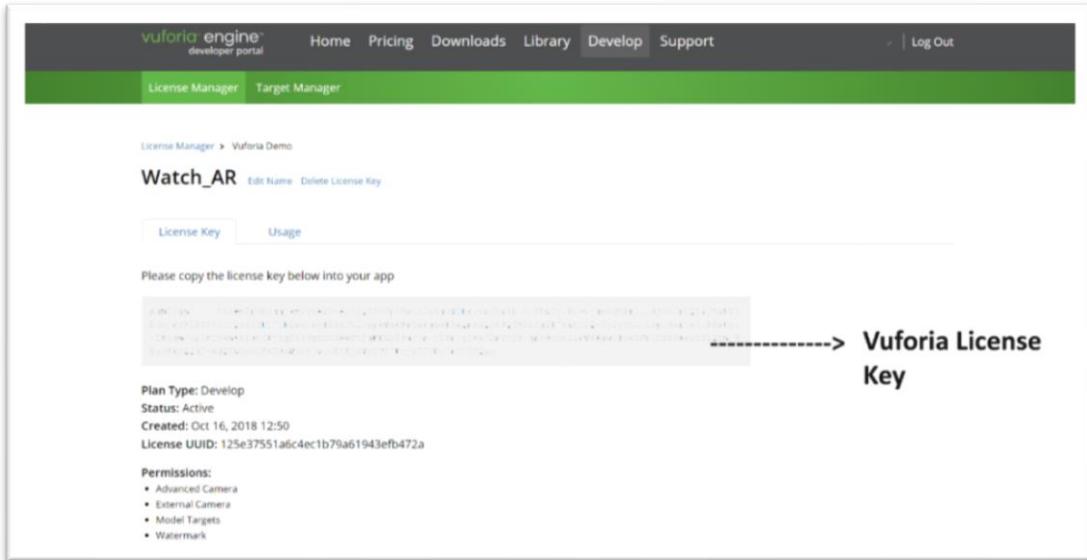


Figure 47: Key Management

The vuforia license provided should match the key mentioned in unity when image recognition process is happening in AR mode.

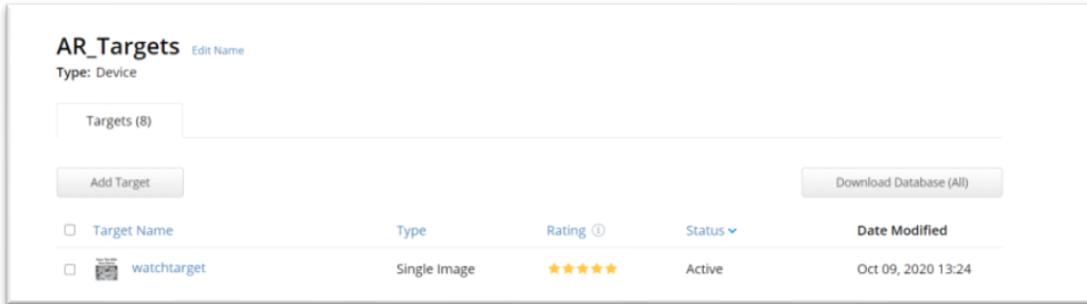


Figure 48: AR_Targets

The AR marker which the user needs to scan should be uploaded in the vuforia cloud. When the user scans the AR marker during runtime this should match with the one which is uploaded in the vuforia cloud. Once this is successful unity will display the respective 3D model of the selected watch in the augmented space.



Figure 49: AR Marker

This image represents the rating provided by vuforia. Higher the rating lesser time will be taken for processing and the 3D model of the selected watch will be displayed in the augmented space.

7.4.2.3. SECURE COMMUNICATION

TLS and Secure Socket Layer (SSL) are cryptographic protocols that ensure data privacy over various communication channels. Since, HTTP data is unencrypted, invalidated, and unverifiable, which allows hackers to spy on user content. We have ensured that a valid SSL certificate is used when data transmission between the app and the server using the HTTPS protocol.

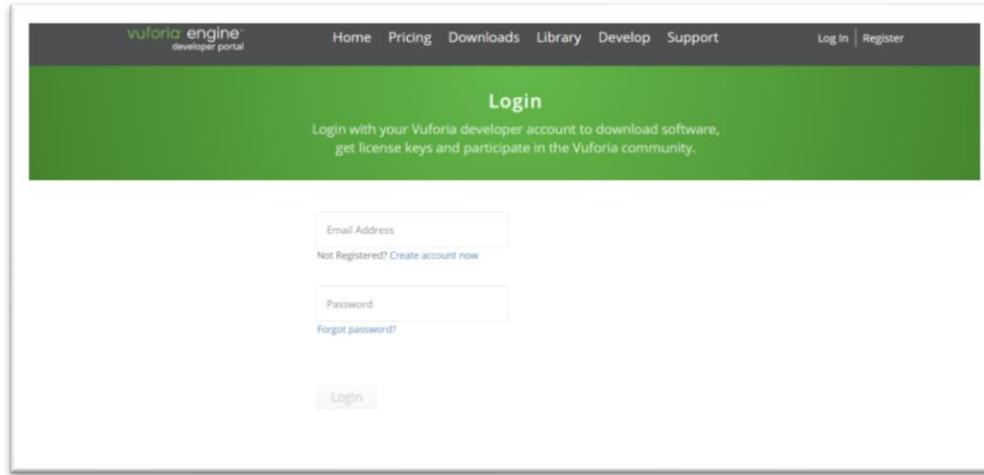


Figure 50: Secure Communication

In order to access the license key provided by vuforia we need to have access to the vuforia database.

7.4.2.4. CACHE ENCRYPTION

The cache is a software component that saves the data temporarily on the user's device. This is used to prevent the delay of data retrieval. Hackers can easily access data stored in cache if it is not encrypted. At times the app does not remove its data after a session ends, and the cache does not expire. If these cache files get into the wrong hands, hackers can manipulate it to access user data or the server. So, we have encrypted cache which is stored in the mobile device stored by the application.

7.4.2.5. CODE OBFUSCATION

This is one of the best ways to protect an app from hacker where they employ code obfuscation techniques. Code obfuscation is an act of creating a code that is difficult for hackers to understand.

```
function myFunc(str) {  
    document.write(str);  
}  
  
var myStr = "My Code";  
myFunc(myStr);
```

(a) original code

```
function msfrt23kjgty(zs12mnjy) {  
    document.write(zs12mnjy);  
}  
  
var nbuqmazsuikh = "My Code";  
msfrt23kjgty(nbuqmazsuikh);
```

(b) obfuscated code

Figure: 5.5 Code Obfuscation

This technique has become popular and is used to conceal code from attacks. Obfuscators are used to automatically convert programming code into a format that cannot be understood by humans. Code obfuscation includes which we have adopted are mentioned below:

- Encrypting some or the entire code
- Removing metadata which may reveal information about the libraries or APIs used
- Renaming classes and variables so they cannot be guessed

Code is obfuscated to prevent data and property from hackers who may reverse-engineer code using software programs.

7.4.3. USER EXPERIENCE TESTING

7.4.3.1. CLIENT - BLACK BOX TESTING

Client had 6 feedbacks after he tested the application without knowing the code.

- a. First one was there were no back buttons to main page,
- b. Second was 4th watch image background looks different from others,
- c. Watch name sizes in below figure are uneven,
- d. Watches are too close to each other,
- e. AR Marker link should be given with button and saved in local memory
- f. Color buttons should be on the button to access easily with left-handed people,

AR Watches

1 no back button

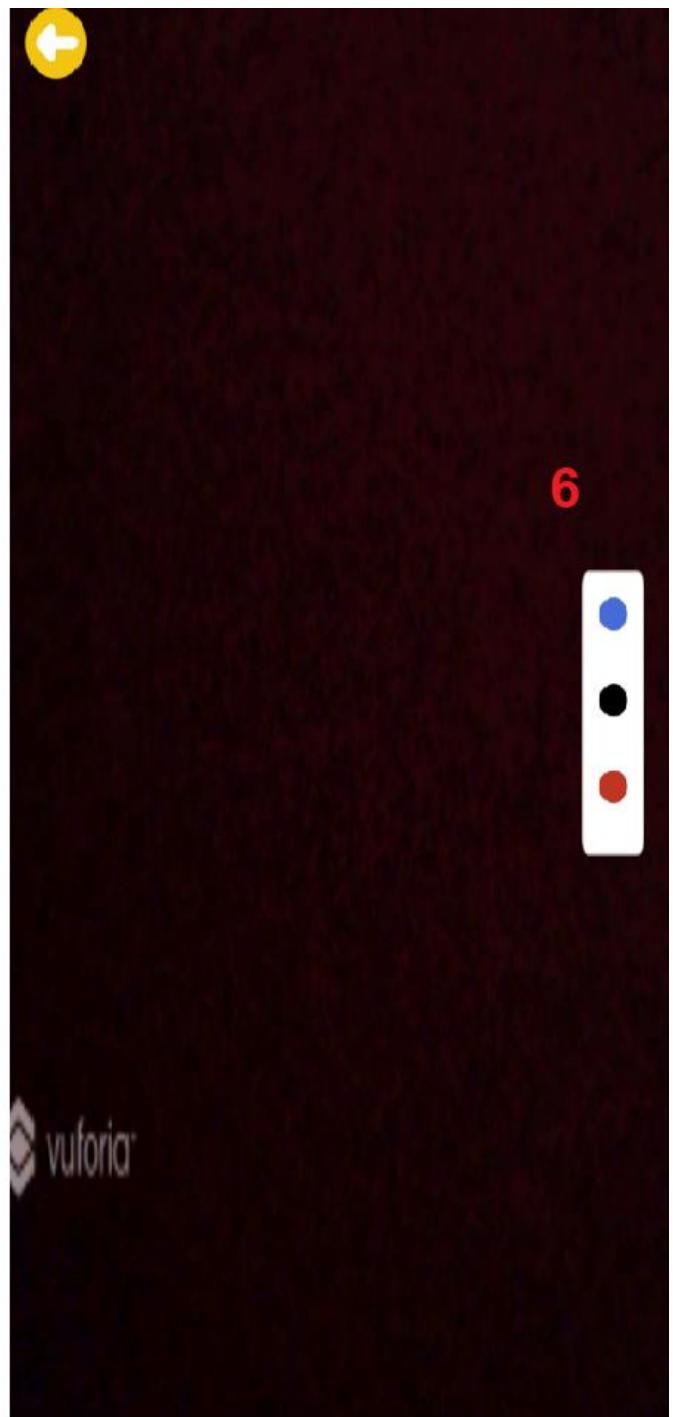


Figure 51: Previous Design

CORRECTIVE ACTION

Software updated according to the feedbacks.

AR Watches



Leather Watch



Smart Watch



Metal Watch



Sports Watch



Press here to get image to see 3D
watch on hand

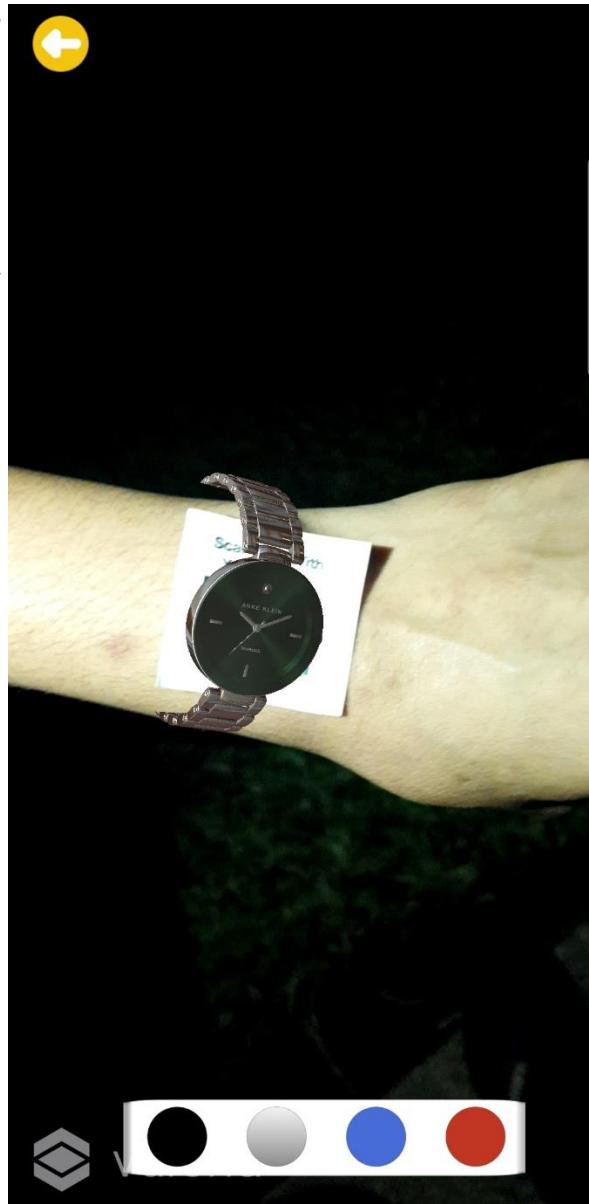


Figure 52: After Corrective Design

7.4.3.2. MODERATED USER TESTING

In moderated user testing 3 people used the application while developer was asking questions checking their interaction with the application.

They found application has a long waiting time after desired watch is selected. In this point application waits 15 seconds to operate camera.

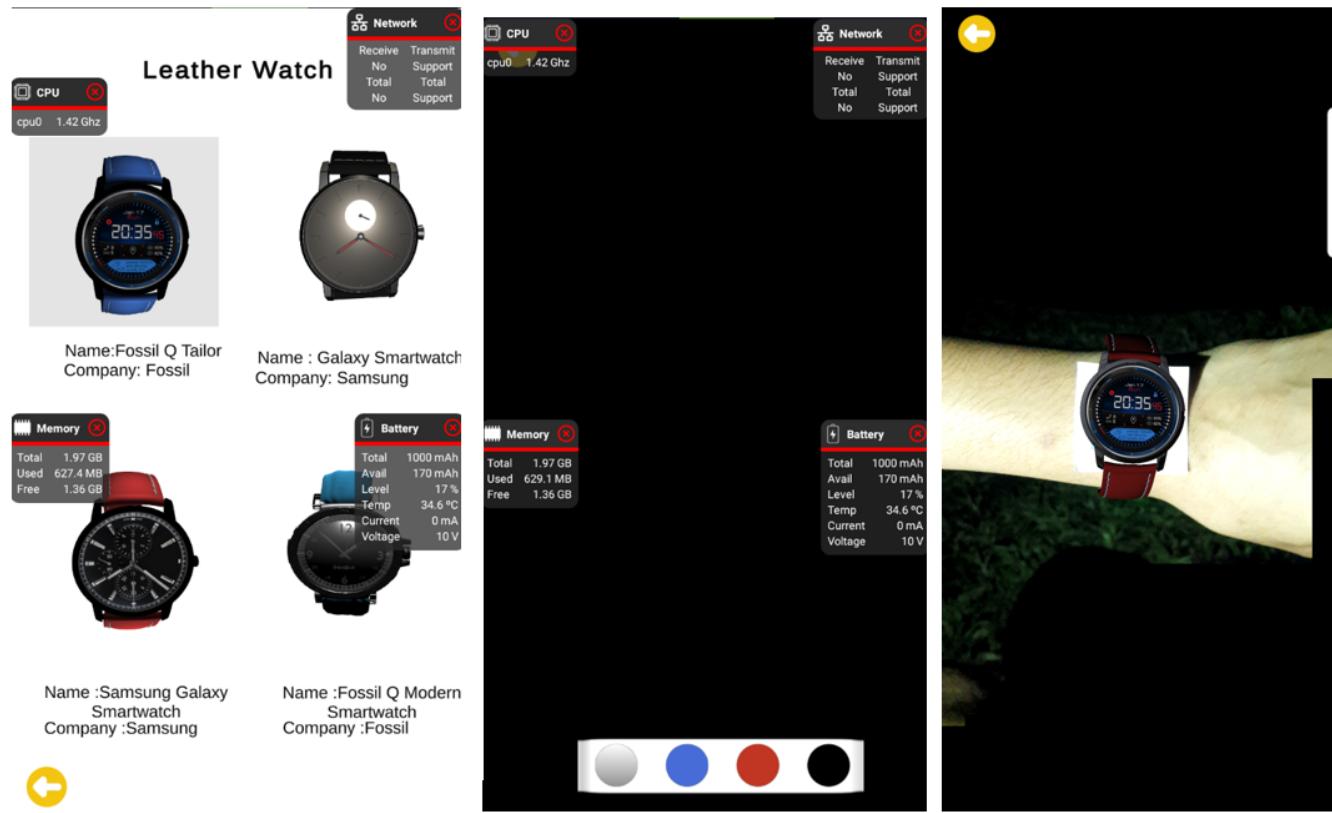


Figure 53: Moderated Test

CORRECTIVE ACTION

However, the time user mentioned about could not reduced since it is related with mobile phone RAM and CPU capacity.

7.4.3.3. UNMODERATED USER TESTING

In unmoderated test, 2 people test the application. One of them was right-handed, other one was left-handed. While left-handed person had no negative feedback, right-handed person told that he had difficulties to go back to previous page when application in on camera mode.

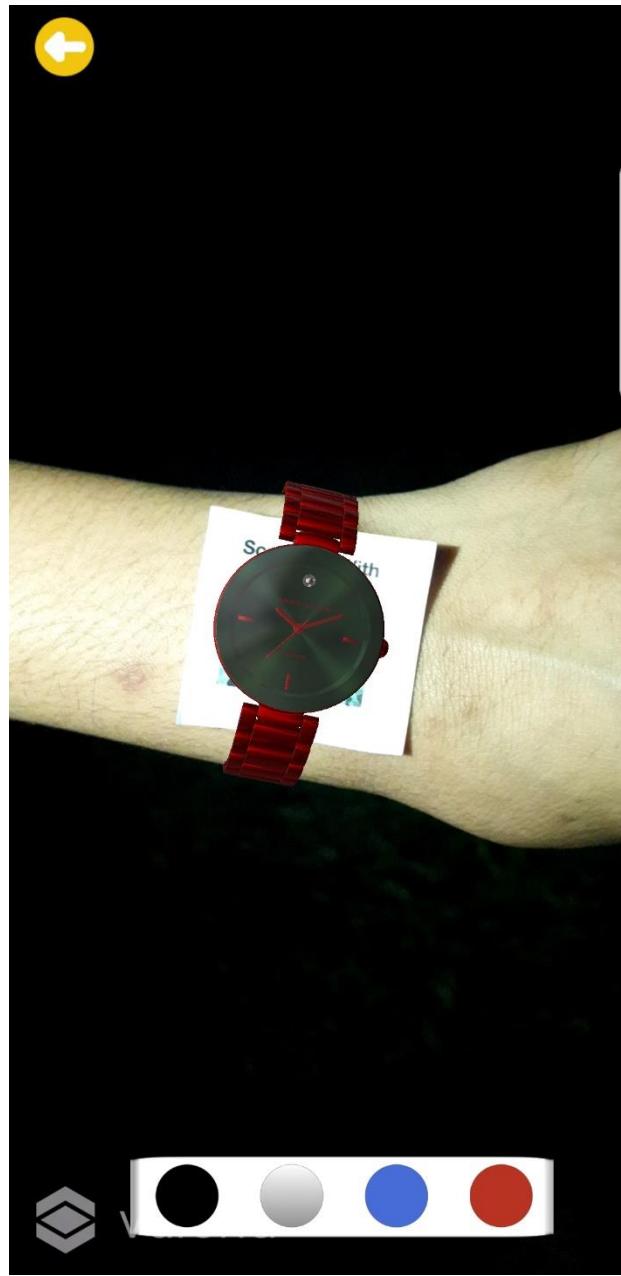


Figure 54: Right-handed Person Could Not Reach the Back Arrow Easily.

CORRECTIVE ACTION

Back button placed to the left bottom corner and software updated.



Figure 55: New Location of Back Button

7.4.3.4. AVAILABILITY AND ACCESSIBILITY

Application is tested 24 hours. No problem is seen and functions are worked as expected.



Figure 56: 24 Hours Test

8. CONCLUSION

In this project, we created AR Mobile Application for every Watch lover which helps them to try watches before buying them. The main objective of the AR Watch project was to solve problems in the market such as Covid-19 and increase accessibility of products to increase sales. This project can help Australian watch shops to increase their sales. For instance, customers who is living in rural areas can buy watches through e-commerce websites without travelling Sydney or other big cities.

These days, Augmented reality trend is skyrocketed. AR can be found in wearable devices, clothes, in-house showrooms, room renting websites, games, etc. It can help create many opportunities in the future.

This application mainly focused on four categories of watches, which are Lather Watch, Smart Watch, Metal Watch and Sport Watch. In application, user can change the straps different colours. During the development, TDD applied. After development, performance, security and user experience tests carried out. After the successful testing of the software it is loaded to android mobile phone and test cases repeated.

Project team understand how to develop Agile and TDD projects. Project team learn new development tool which is Unity Software. Unity platform is used for creating and designing the application. Throughout the project, software processes, project management tools and professional ethic rules applied to the project.

FUTURE ENHANCEMENT

-  We will do buy feature for the app for online shopping
-  Fix the watches to different hand sizes
-  Try to reduce complete watches by separating dial and strap and user selecting and making a watch

Figure 57: Future Enhancement

8.1. LESSON LEARNED

Date	Description of problem/opportunity	Recommended Action for next time/project	Lesson Learned
09.08.2020	Lack of communication plan	Developing a good communication plan between team members. Well explained procedures and policies for whom works individually or remotely.	Team communication is important. Some actions might affect all team members even though they are taken just for one person.
19.09.2020	Project Time Management	Timely managed task are important to finish project on time.	Team had overworked because of lack of Project Time Management

9. BIBLIOGRAPHY

1. Weston R. 2017, Serverless Architectures and Continuous Delivery, <https://www.gocd.org/2017/06/26/serverless-architecture-continuous-delivery/>
2. Esplin, C 2016, What is Firebase, Medium, How to Firebase.
3. Gergely, T, Balogh, G, Horváth, F, Vancsics, B, Beszédes, Á & Gyimóthy, T 2018, 'Differences between a static and a dynamic test-to-code traceability recovery method', *Software Quality Journal*, vol. 27, no. 2, pp. 797–822.
4. Huluka, D & Popov, O 2012, Root cause analysis of session management and broken authentication vulnerabilities, *IEEE Xplore*, pp. 82–86, viewed 27 May 2020, <<https://ieeexplore.ieee.org/abstract/document/6280203>>.
5. L. Parody 2018, How to Manage Modern Software Projects: Waterfall vs. Agile <https://medium.com/@lizparody/waterfall-vs-agile-methodology-in-software-development-1e19ef168cf6>
6. Sommerville, F. I. 2016, Software Engineering, 10th Edition, Pearson Education © 2016.