NOVUS

ZEN CODING

"THE BID"

DETAILS AND SUMMARIZES USER REQUIREMENT

RISK ANALYSIS FOR NOVUS

PROJECT DECISION AND PLAN

PROJECT COSTING

USABILITY EVALUATION OF MOCK-UPS

VS, IE, KD, ER, AL, AK, MZ, AD

Group 3

20/10/2020

TABLE OF CONTENTS

REQUIREMENTS

RISK ANALYSIS

PROJECT DECISIONS AND PLAN

PROJECT COSTING

USABILITY REPORT

DOCUMENT DETAILS AND HISTORY

1. REQUIREMENTS

1.1 Introduction

1.1.1 Purpose

The document describes the requirements for the social photo sharing system as requested by HWU. This system is intended to be used by members of the general population to share and interact with photos of objects and animals. The user will be able to post their own photos and edit them whilst also interacting with other users' posted photos creating a social community based around the sharing of these photos.

This document is created for the benefit of the client and Zen Coding.

1.1.2 Scope

The system will be both compatible as a mobile application and as a web application to ensure full compatibility across all devices. The system will use a variety of frameworks and technologies to ensure both the full functionality of the system and the creating of a easy to use interface for the user.

The databases will hold:

1. Information about the user's profile (eg. username, password, email, etc.).
2. The user's uploaded photos and their relevant meta-data (eg. GPS data).
3. Each photo's "community" aspects (eg. comments, points, etc).
4. Information about each user's usage of the app (eg. time spent on app, most searched terms, etc.)

The general purpose of the system will be for users to share and create communities around pictures of objects and animals. Users should have the ability to express themselves creatively on the system and connect with other users to create a social network.

Furthermore, user data will be sent back to administrators where many facets of it can be studied and analyzed (eg. time spent on app, user interests, user age, photo point rankings, etc.)

1.1.3 Overview

This document will outline a general description of the system but more importantly will outline in detail specific functional and non-functional requirements of the system developed by Zen Coding for the client.

1.2 General Description

1.2.1 Product Perspective

The product is being developed for the client and will have similar capabilities to social networking applications such as Instagram which focus on the sharing of photos. The product is designed with the emphasis on sharing photos of animals by the use of individuals.

1.2.2 General Capabilities

The main functional requirements are:

- Allow users to upload photos
- Allow moderators to modify or remove content
- Allow users to interact with another user's photo
- Analytical data is collected and sent to administrators

In essence, the product would allow users to post their own photos and interact with other users photos. Our vision for the product is for it to become a lively and vibrant social networking platform based upon the sharing of animal photos.

Allow users to upload photos

The product should allow users to upload photos to their own personal page. The uploaded photo can be taken from within the app or from outside. Furthermore, users should have the capacity to edit these uploaded from within the application.

Allow moderators to modify or remove content

Moderators of the product should be allowed to remove content as seen fit. Certain content is prohibited from the product (such as photos of people) and moderators should have the capability to remove said content.

Allow users to interact with another user's photo

Users will have the capacity to interact with another user's photo. This may be through the comments system or through the points system. In addition, a user will be able to see the results of their interaction through global displays such as competitions or leaderboards.

Analytical data is collected and sent to administrators

Analytical data such as number of hours spent on the product, most popular tags, number of photos posted, number of points, etc. will be collected and sent to an administrative body for data analysis.

1.2.3 General Constraints

- The product must be compatible on PC and mobile
- Product interface must be user friendly (ie. Easy to understand and appealing)
- User data must adhere to GDPR standards
- The product must be well optimized and run fast on all devices
- The systems database capacity must be scalable
- Normal user permissions and power will be different to a moderator’s
- Product to be complete by the first of May 2021

1.2.4 User Characteristics

The large majority of users are expected to be regular individuals. The products are meant to appeal to a wide audience of individuals who are passionate about animals and who intend to use the product as a means of social networking with others who feel the same way.

Furthermore, we do expect a small presence of organizations within animal industries to use the application. This may be for advertising purposes or for the betterment of public relations.

1.2.5 Operational Environment

The product will be compatible on all internet browsers, Android and IOS.

Flutter will be used to create the app. Flutter is an SDK by Google that allows for the development of apps which are compatible with multiple platforms using the same codebase.

Firebase will also be primarily used as the database for the product. This is because it offers interoperability with flutter and will allow us to store and use all relevant data in a comprehensive and simple manner.

1.3 General Requirements

1.3.1 Capability Requirements

High Priority 🔴

Medium Priority 🟡

Low Priority 🟢

F-UR1 User Interaction

F-UR1-1 🔴 Comment Section

Every photo must have its own comments section.

F-UR1-2 🔴 Comment Manipulation

Users must have the ability to post/delete and edit comments on a photo comment section.

F-UR1-3 🟡 Point Section

Users should be able to “score” a photo using the point system.

F-UR1-4 🔴 Channel Creation

Users must create channels which act as the user’s hub for all their photos.

F-UR1-5 🟡 Photo Search

Users should have the ability to search for photos based on photo tags and other relevant information.

F-UR1-6 🔴 User Following

Users must be able to follow other users which means they are more likely to see their photos in their feed.

F-UR1-7 🟢 Photo GPS Search

Users could search for photos based off the location the photo was taken.

F-UR2 User Analytics

F-UR2-1 🟡 Leaderboard of Photos

There should be a leaderboard of all top-rated photos.

F-UR2-2 🔴 Summary Analytics Report

A report must be generated which supplies administrator with the current state of the platform.

F-UR3 Photo Manipulation

F-UR3-1 🔴 Photo Upload

Users must have the ability to upload photos to the platform.

F-UR3-2 🟡 Photo Tagging

Users should have the ability to tag photos to categorize them.

F-UR3-3 🟡 Photo Editing

Users should be able to make changes and edits to their photos in the app.

F-UR3-4 🟡 In-app Camera

Users should be able to take photos from inside the app.

F-UR3-5 🟢 Location Tagging

Users could tag the photos they have taken with its relevant GPS metadata.

F-UR4 Photo contest

F-UR4-1 🔴 Create contest

Users must be able to create general or theme-based contests.

F-UR4-2 🔴 Contest settings

Creator of the contest must be able to edit contest settings based in their preference.

F-UR4-3 🔴 Join contest

Users must be able to participate in different contests.

F-UR4-4 🔴 Leave contest

Users must be able to leave contests.

F-UR4-5 🟡 Moderate the contest

Creator of a given contest should have full moderation control.

F-UR5 Leaderboard

F-UR5-1 🔴 Data display

Data from different contests must be displayed

F-UR5-2 🟢 Trending

Trending contests could be displayed.

F-UR5-3 🟡 Player username

Username of the players with the points should be displayed.

F-UR6 Group

F-UR6-1 🔴 Create group

User must be able to create a general or theme-based group

F-UR6-2 🔴 Join group

Users must be able to join different groups

F-UR6-3 🔴 Leave group

Users must be able to leave groups.

F-UR6-4 🔴 Group settings

Creator of the group must be able to edit the settings of the group

F-UR6-5 🟡 Moderation the group

Creator of the group should have the ability to moderate the content of the group

F-UR7 Registration

F-UR7-1.1 🔴 Register with Email

User must be able to sign up using an email address and password.

F-UR7-1.2 🟢 Verify Email

User could receive email upon registering.

F-UR7-2.1 🔴 User Login to account

User must be able to sign in as a recurring user with their email and password.

F-UR7-2.2 🔴 User Logout of account

User must be able to logout of their account.

F-UR7-3 🔴 Secure login credentials

User details must be stored securely in the system.

F-UR7-4 🔴 Deletion of Account by User

User profiles must be removed from the system once deleted by the user.

F-UR7-5 🟡 User adds profile details upon account creation

User should be able to add their profile details when creating their account.

F-UR8 Profile

F-UR8-1 🔴 Profile Details

User must have the capability to edit profile details such as the display name, DOB, profile picture and description.

F-UR8-2 🟢 Archive Posts

Posted deleted by the user could remain in a user archive for them to refer to.

F-UR8-3 🟡 Follower/Following Count

Following and follower count should be displayed on a profile.

F-UR8-4 🟡 User Posts

All post associated to a profile should be displayed when a user views another or their own profile.

F-UR8-5 🟡 Profile Searching

Users should search for other users on the platform.

F-UR9 Administration

F-UR9-1 🔴 Administration support

Administration tools must be available to manage accounts and data.

F-UR9-2 🔴 Delete Account

Admins must be able to delete accounts of users.

F-UR9-3 🔴 Edit Account

Admins must be able to edit a given users account.

F-UR9-4 🔴 Maintain system stability

Admins must be able to handle bug reports or issues while the system is active.

F-UR3-5 🟡 Moderate the system

Admins should be able to moderate content as well as users on the system

1.3.2 Constraint Requirements

High Priority🔴

Medium Priority 🟡

Low Priority 🟢

NF-UR1 Hardware

NF-UR1-1 🟡 Environment

The system should be hosted on an internet-connected web server.

NF-UR1-2 🟢 System Optimization

The system could be optimized to run smoothly on various devices using faster/slower hardware.

NF-UR1-3 🟢 Storage Scalability

The systems database capacity could be scalable in order to account for increasing users and data on the platform.

NF-UR2 Software

NF-UR2-1 🟡 Viewports

There interface of the system should have different viewports based on different devices such as mobile, tablet and desktop.

NF-UR2-2 🔴 Browser Support

The website for the system must run across all widely supported web browsers.

NF-UR2-3 🟢 Disability Support

The system could provide functionality to improve accessibility for disabled users (deaf, colorblind, etc).

NF-UR3 Security

NF-UR3-1 🔴 Appropriate view of data

Users must be able to access and edit their own data and not someone else’s.

NF-UR3-2 🔴 Credentials Stored Securely

Users credentials must be stored securely in the system

NF-UR3-3 🔴 Security techniques

The system must incorporate standard techniques to prevent code injection which could break the system.

NF-UR3-4 🟢 Validate Environmentalists

Environmentalists registering to the system could be validated by an administrator before they can start posting.

NF-UR3-5 🟡 Privacy Options

Users on the system should have the option to hide their content from people that aren’t following their account with their knowledge.

NF-UR3-6 🟡 Privacy Options

Users on the system should have the option to hide their content from people that aren’t following their account with their knowledge.

NF-UR3-7 🟢 Criteria for password

Users registering to the system could follow a set criterion for their passwords, which could be mix between numbers, letters, symbols.

NF-UR4 Time

NF-UR4-1 🔴 Deployment Deadline

The system must be ready for use by the 1st April 2021.

NF-UR5 Performance

NF-UR5-1 🟡 Responsiveness

The system should respond to user actions within 2 seconds.

NF-UR5-2 🟡 Large active user base

The system should handle over 100 users at a given moment.

NF-UR5-2 🟡 Fast read/write database speeds

The systems database should have fast reading or writing speeds for data

NF-UR6 Data

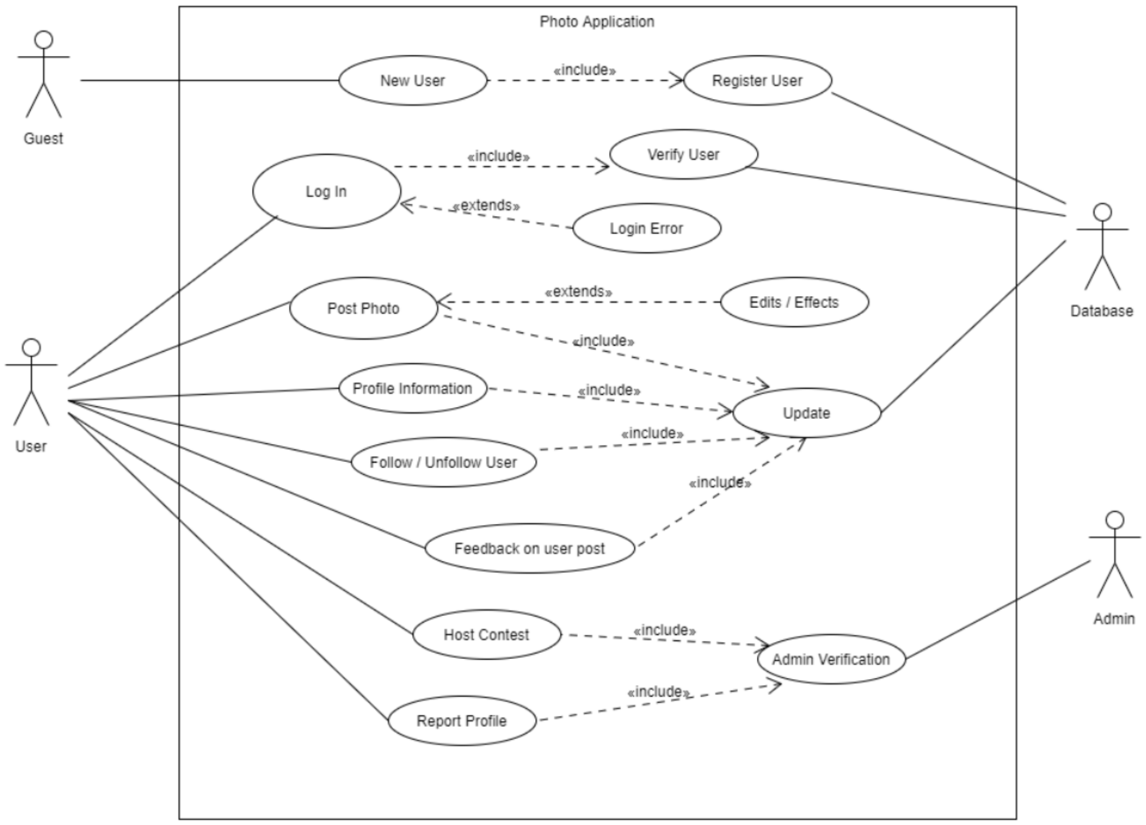
NF-UR6-1 🔴 Storage of data

All user data on the system must be stored on firebase which would have a limited size.

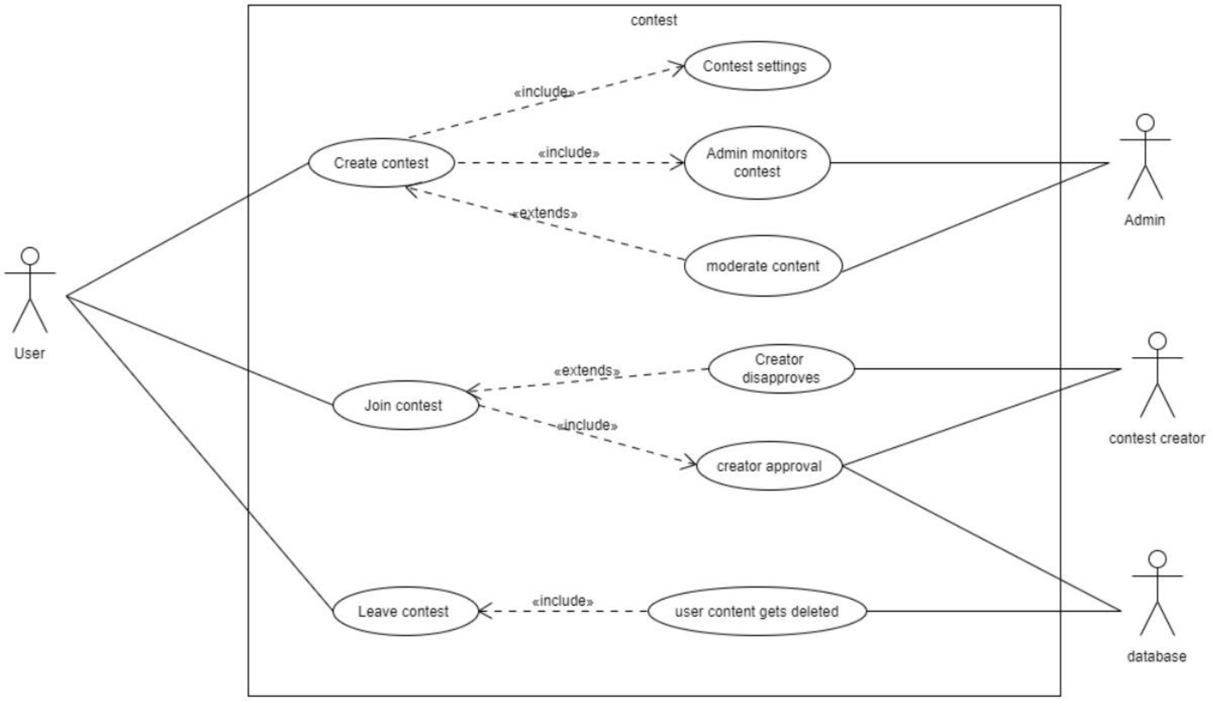
1.4 Appendices

1.4.1 Use Case diagrams and Use Case specifications

Use Case Diagram 1



Use Case Diagram 2



Use Case Specifications

USECASE: Login to Account

DIAGRAM: Diagram 1

GOAL: Access to the System

ACTORS: User

- Main Success Scenarios:
  1. User launches application
  2. Enter account information
  3. Verify user on platform
  4. Confirmation of login
- Extensions:

  3a. User account doesn't exist
    1. Display login error
    2. User may re-enter account details
    3. Return to MSS step 3

USECASE: Post Photo

DIAGRAM: Diagram 1

GOAL: Photo on the system

ACTORS: User, Database

- Main Success Scenarios:
  1. Captures photograph
  2. Provide details for the post
  3. Confirms and Post to account
  4. Update system
- Extensions:

  2a. Add edits/effects to photo
    1. User modifies the image
    2. Return to MSS step 3

USECASE: Host Contest

DIAGRAM: Diagram 1

GOAL: Competition available

ACTORS: User, Administrator, Database

- Main Success Scenarios:
  1. User decides to host an event
  2. Admin verifies legitimacy of the event
  3. Event published to public
  4. System updated
- Extensions:

  2a. admin denies the event
    1. Admin disallows the event to be created
    2. User is notified of cancelation

USECASE: User leaves feedback on a post

DIAGRAM: Diagram 1

GOAL: Update system with post information

ACTORS: User, Database

- Main Success Scenarios:
  1. User leaves feedback as like/comment
  2. Confirms and the like/comment gets added
  3. Database is updated with new information about the post
  4. New data is visible for public to see on the post

USECASE: User reports account on the system

DIAGRAM: Diagram 1

GOAL: Remove reported user from the system when there is valid evidence

ACTORS: User, Administrator, Database

- Main Success Scenario:
  1. User leaves report claim on user
  2. Administrator reads the claim
  3. Administrator confirms claim and removes user from the system
  4. Database gets updated
- Extensions:

  3a. Rejected claim
    1. Administrator rejects the claim due to insufficient evidence
    2. User is notified

USECASE: Create contest

DIAGRAM: Diagram 2

GOAL: Contest available to the other users

ACTORS: Primary actor: User

Secondary actor: Admin

- Main Success Scenarios:
  1. User creates a contest
  2. User sets/edits the contest on personal preference
  3. Contest starts to take place and is presented to other users
  4. Admin monitors the contest
- Extensions:

  3a. the user/admin wants to make changes to the content uploaded
    1. Disables the contest for some time
    2. Moderates the content posted by the users
    3. Return to MSS step 2

USECASE: Join Contest

DIAGRAM: Diagram 2

GOAL: User can upload photos to the contest.

ACTORS: Primary actor: User

Secondary actor: contest creator, database

- Main Success Scenarios:
    1. User finds the contest theme they want to join.
    2. User sends to request to the contest creator.
    3. Contest creator approves the user.
    4. User can start to post their content.
    5. Database gets updated
- Extensions:

  3a. Contest creator denies approval
    1. Send message of denial to the user.
    2. Return to MSS step 1

USECASE: Leave contest

GOAL: User leaves the contest, and all his content gets deleted.

ACTORS: Primary actor: User

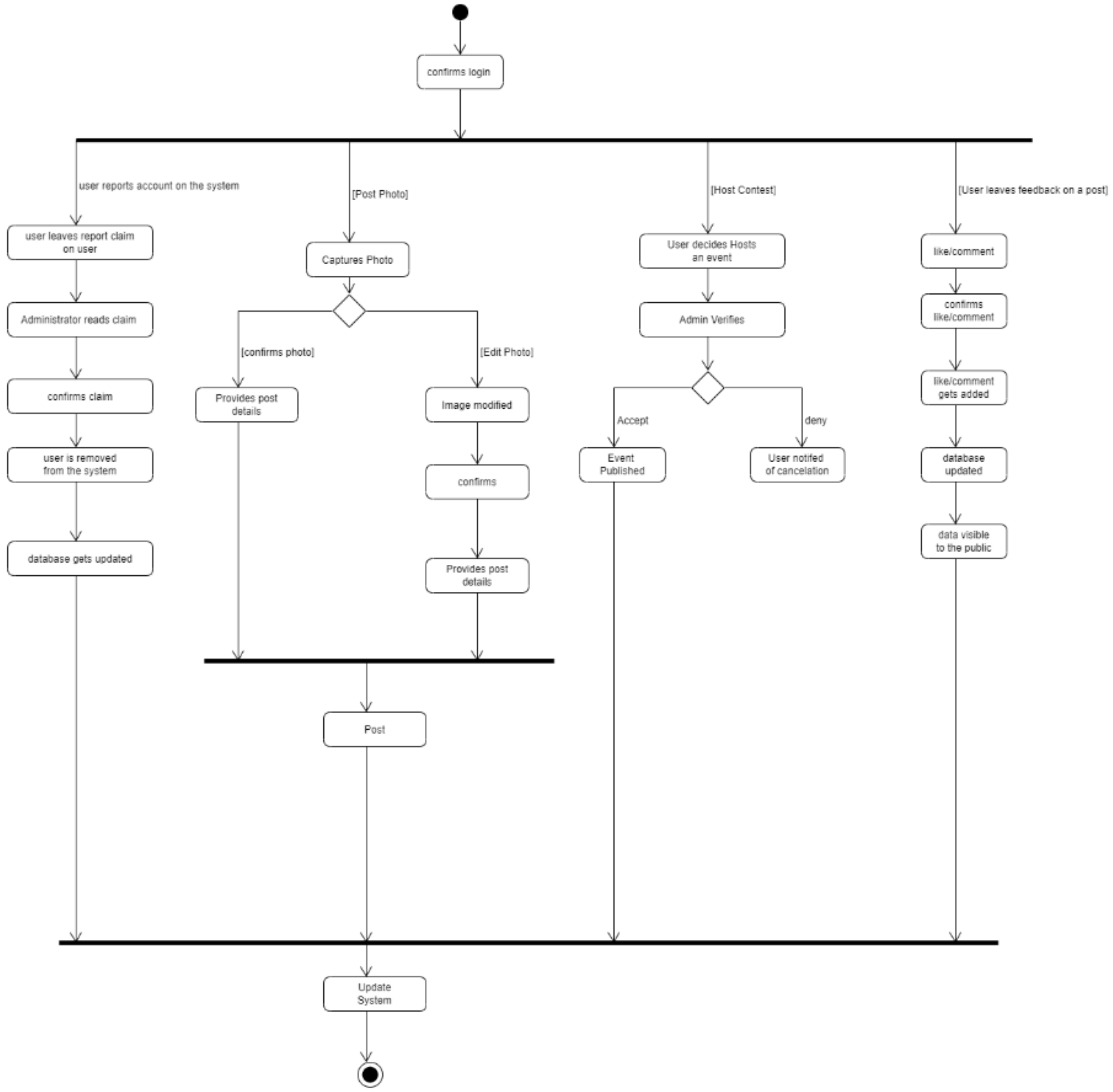
Secondary actor: Database

Main Success Scenarios:

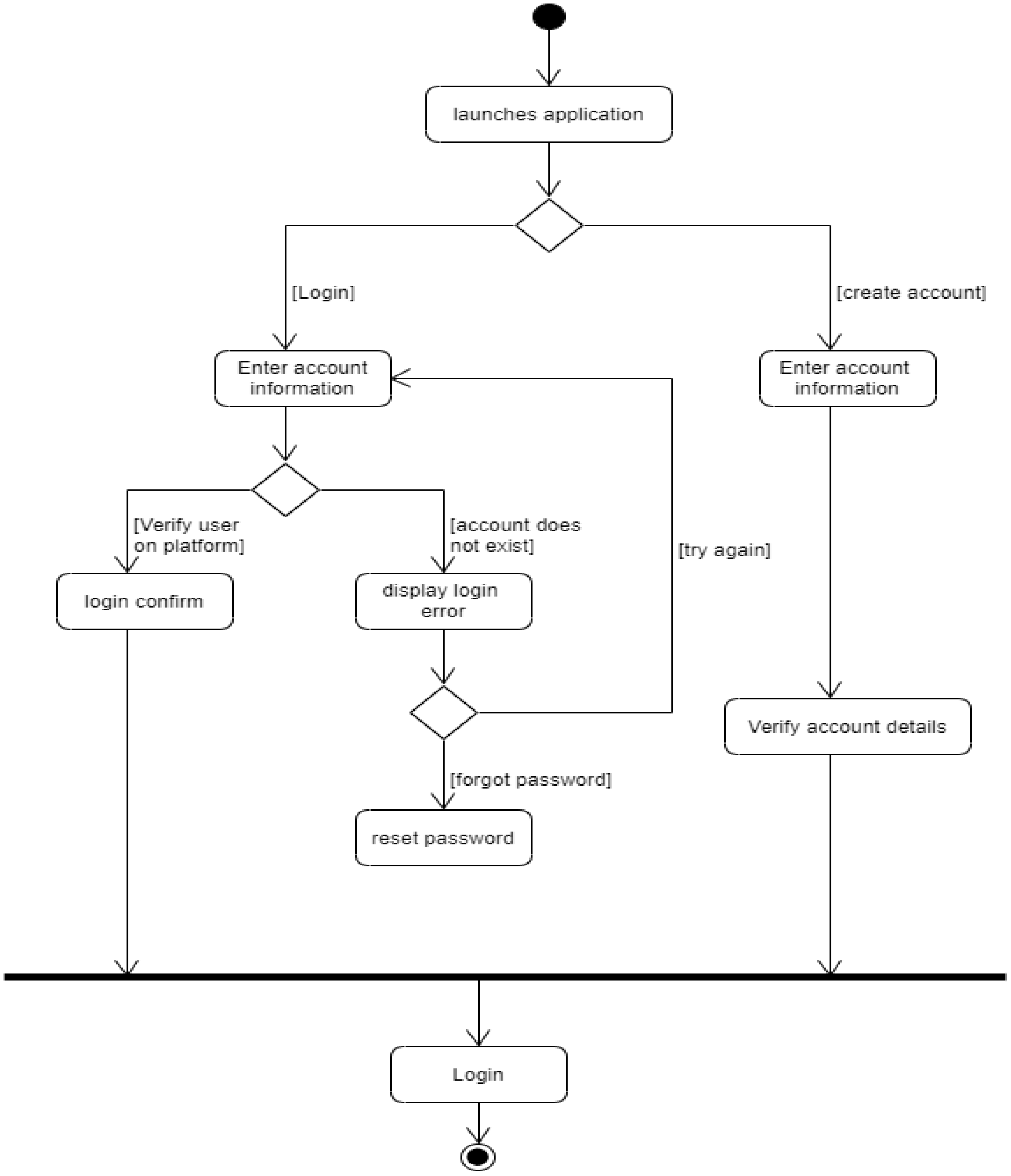
1. User leaves the contest
2. All content posted by the user gets deleted
3. Database gets updated

1.4.2 Activity Diagrams

Activity Diagram – General actions that a logged in user can perform

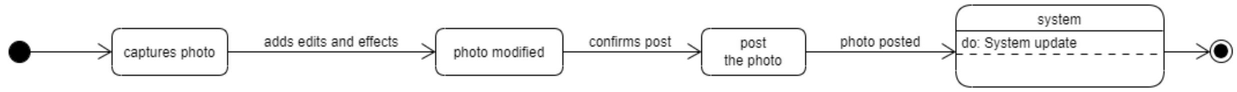


Activity Diagram – A new user or someone logging back in

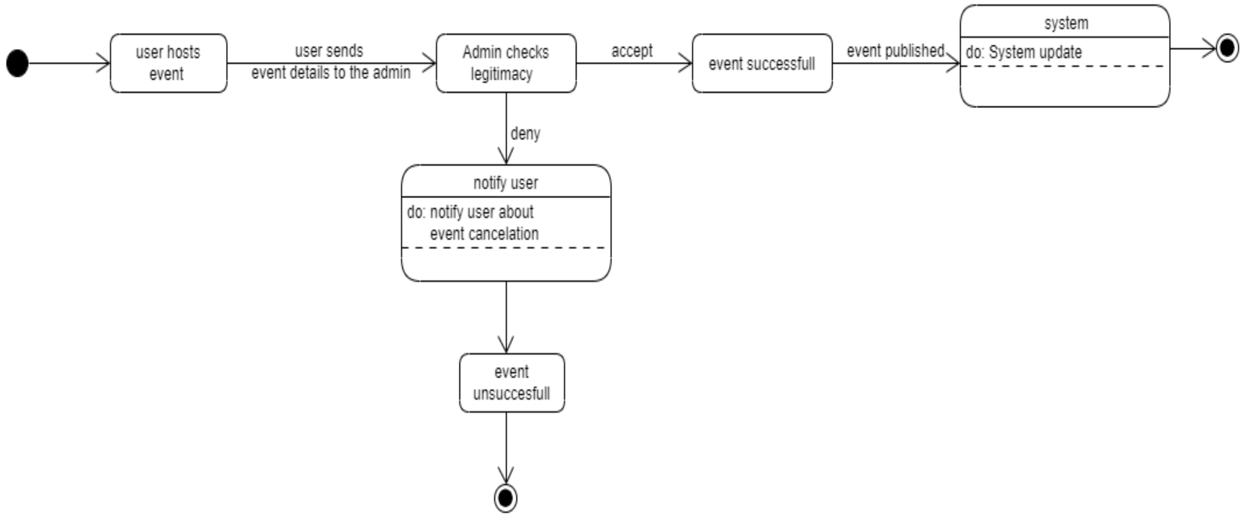


1.4.3 State Diagrams

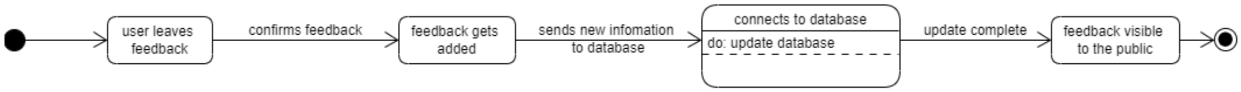
State Diagram – process of photos getting uploaded



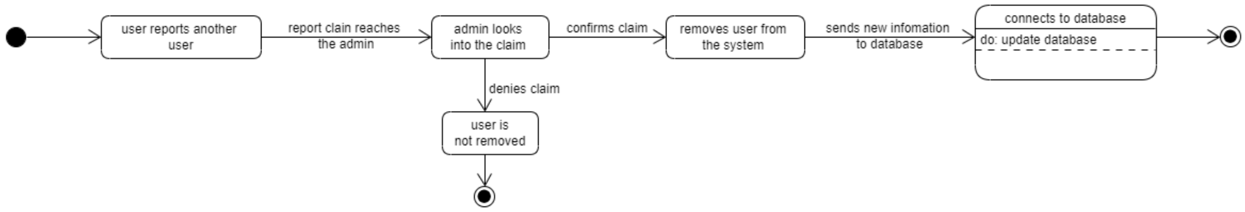
State Diagram – process for users hosting a contest



State Diagram – process of user leaving feedback on a post



State Diagram – process of user leaving feedback on a post



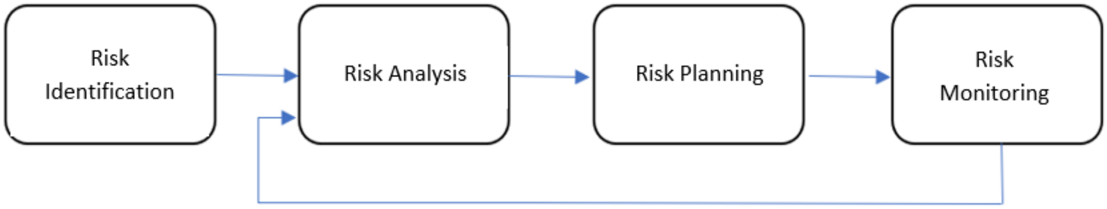
2. RISK ANALYSIS

2.1 Risk Management Approach

2.1.1 Background

A risk management plan is generated to minimize the impact of any potential risks that may occur during the project. The risk management strategy can be

broken down into four key stages that take place sequentially.



1. *Risk Identification*

   At this stage, we identify and list out the potential risks as product risks, project risks and business risks.

   i. *Product risks:*
   These risks have an impact on the quality and performance of the project.

   ii. *Project risks:*
   These risks have an impact on project resources and deadlines.

   iii. *Business risks*
   These risks have an impact on the company creating or acquiring the project.

2. *Risk Analysis*
   At this stage, we access the list of risks and analyze them. We figure out the likelihood, the consequences and effect of each risk.

3. *Risk Planning*

   At this stage, we draw up a plan to prevent and mitigate the impact of the risks.

4. *Risk Monitoring*
   At this stage, we monitor the risks as we go ahead with our project. Then, we add on to our risk analysis as more information about the risks is acquired.

2.2 Risk Introduction

2.2.1 Background

The initial stage is the most important stage of the whole risk analysis part, that is Risk Identification. The second stage is to calculate the impact of these risks and the probability of them occurring. These priorities are labelled using the colors mentioned in the tables in the documentation part. The third and final stage is the remedial action as to how to avoid this risk from happening, and the monitoring of these risks, which includes how to avoid/minimize it and the steps that should be taken once occurred.

These pre-defined risk types provide a good base that helps to identify the risk and ensures that a certain process is followed to deal with the mentioned risks.

After identifying and categorizing a risk, it is documented.

i. *Staff Risks*
Risks that are related to the members of the Group involved in the project.

ii. *Technical Risks*
Risks that are related to the software and hardware aspect/limitations of the group project.

iii. *Accessibility Risks*
Risks that are linked to the availability of the code for the various group members.

iv. *Organizational Risks*
Risks that emerge from the organizational environment including the commerce side of things where the software is made and where its used.

v. *Estimation Risks*
Risks that are derived from management of time, i.e., the parts of the project and the project itself to be completed on time.

vi. *Security Risks*
Risks that are related to the security of the code and the threats of it being attacked by viruses, along with the theft of the software/code.

2.2.2 Documentation

Low Priority 🟢

Medium Priority 🟡

High Priority 🔴

Risk ID	Risk Type	Risk	Probability	Impact	Remedial Action
RID1	Staff 🟡	Lack of group arrangement	Moderate	Tolerable	Any meetings should be scheduled in advance AND with every group member's say taken in account. Roles and tasks should be clear to anyone who has been assigned one.
RID2	Technical 🟡	Code unclear to some	High	Tolerable	Any piece of code written down should have a reasonable amount of comments written to avoid confusion from within the group members.
RID3	Staff 🟡	Poor communication	High	Serious	Group members should be notified whenever there is any change in plans or objectives regarding the project.
RID4	Technical 🟡	Unexpected outcomes and errors	High	Serious	Functions should be implemented and tested in singles to allow for easier tracking of error outcomes.

RID5	Technical 🟡	Software limitations	Moderate	Serious	Code should be written in a way such that it allows for future expansion for extra features.
RID6	Accessibility 🟢	Lack of accessibility	Low	Tolerable	Anything involving this project should be posted in a place accessible and possibly updateable by all group members of the group, possibly GitHub.
RID7	Technical 🟢	Loss of content/information	Moderate	Tolerable	Keep all files organized and double-check before deleting work.
RID8	Technical 🔴	Most of functional and non-functional requirements are not met.	Low	Serious	Team members can keep track of all the requirements as they go ahead with the project. One of the members can create a checklist of all the requirements that have been met.
RID9	Staff 🔴	Inconsistent costumers	Moderate	Serious	Have regular meetings with the customer, thereby we get a clear assurance of what is to be implemented. Providing them with prototype and progress at every stage of the project.
RID10	Organizational 🟡	Financial problems	Moderate	Serious	Project costing should be done realistically and cleverly. Once the budget is fixed, it should not be exceeded.
RID11	Staff/ Technical 🔴	Team members may not have sufficient knowledge to do certain aspects of the project.	High	Serious	Assign tasks to the members based on their strengths and team members who are familiar with the topics should help the rest overcome obstacles.

RID12	Technical 🔴	Server failure	Moderate	Serious	If the server fails for any reason whatsoever, there is a high risk of losing valuable data. We can avoid that by using incremental data backups, which work on backing up data only when data is added or modified and adds that to the previous backup.
RID13	Technical 🔴	Ill-fitted photos posted	Moderate	Serious	Constant monitoring to make sure any unsuitable photos do not exist on the feed, consequently giving out warnings and possibly temporary account limitations to users who belong to that criteria.
RID14	Estimation 🟢	The project targets not met by a certain time	Moderate	Serious	We should set realistic timings as for when goals are to be met. We should observe our previous behavior to learn from any mistakes or bad decisions made, and possibly prioritize our timings better for more future productivity relative to the project.
RID15	Security 🔴	DDOS attacks	Low	Serious	DDOS attacks cannot be predicted, but we can reduce the chance of it

					happening by having someone monitor the firewall for any suspicious activity.
RID16	Staff 🟢	Team lacks motivation	Moderate	Tolerable	The team members should constantly motivate each other.
R1D17	Organizational 🔴	Illegal documentation	Low	Serious	Team members should access only legal documents.
R1D18	Technical 🟢	User experience is poor and User interface is not aesthetically pleasing.	Moderate	Tolerable	Assign UI/UX to the designers in the group. Polls may be put up to see which interface Staff prefer more.
R1D19	Organizational 🔴	Unable to raise extra capital if cost exceeds budget.	Low	Serious	The group members must try to raise money for the project via various sources such as private/government banks, individual asset lenders.
R1D20	Organizational 🟢	Copyright/Patent issues.	Low	Tolerable	Sometimes when the final product is done and dusted, there are issues with the copyright/patent where someone has already made a similar product, or the name of our project conflicts with other projects.
R1D22	Staff 🟡	Team members can have problems with their workload	High	Tolerable	Group members should let others know if they are having problems with their workloads as this can cause failure to produce work. Hence work needs to be split amongst members equally for productivity.

2.3. Risk Planning

2.3.1 Background

Once the risks have been identified, we must find the appropriate risk management methods.

2.3.2 Risk Strategies

There are three types of risk strategies we must follow, and they are: -

1. Avoidance:
   The avoidance strategy reduces the probability of occurrence of the risks.

2. Contingence:
   The contingency strategy is assuming the worst case and developing a plan to deal with it.

3. Minimization:
   The minimization strategy is intended to reduce the impact of the risks.

2.3.3 Documentation

Risk (RID1): Lack of group arrangement

1. Avoidance: Any meetings should be scheduled in advance AND with every group member's say taken in account. Roles and tasks should be clear to anyone who has been assigned one.

2. Contingence: In case the group lacks total order and organization, a meeting should be conducted for the sole purpose of discussing this matter, to try to work around this obstacle.

3. Minimization: Group members should always update each other on edits they add to the project. Communication should be constant.

Risk (RID2): Code unclear to some

1. Avoidance: Any piece of code written down should have a reasonable amount of comments written to avoid confusion from within the group members.

2. Contingence: In case code is vague to group member(s), a meeting should be conducted to make sure all coding doubts are cleared and fully explained.

3. Minimization: Code comments should be updated frequently, especially when new code is added.

Risk (RID3): Poor communication

1. Avoidance: Group members should be notified whenever there is any change in plans or objectives regarding the project.

2. Contingence: Nothing too complicated here, if communication is weak, a meeting should be scheduled to discuss the importance of communication in this project and how strongly it affects it.

3. Minimization: Meetings should be routinely scheduled.

Risk (RID4): Unexpected outcomes and errors

1. Avoidance: Functions should be implemented and tested in singles to allow for easier tracking of error outcomes.

2. Contingence: In case we encounter errors deeper into the app, we should trace the error to know its exact source. However, we hope that this is not the case in the future.

3. Minimization: Make sure every function works as intended and testing it immediately after it is done.

Risk (RID5): Software limitations

1. Avoidance: Code should be written in a way such that it allows for future expansion for extra features.

2. Contingence: In case we encounter a wall midst coding, we should try our best to modify certain aspects of our whole code to allow for more functions to work.

3. Minimization: Always code in a way that allows for countless features to be implemented in the future.

Risk (RID6): Lack of accessibility

1. Avoidance: Anything involving this project should be posted in a place accessible and possibly updateable by all group members of the group, possibly GitHub.

2. Contingence: Any member that encounters problems when trying to access the project files should immediately report to the group so that he/she can access them.

3. Minimization: Always push commits for any changes or modifications made to the project files and keep it updated.

Risk (RID7): Loss of content/information

1. Avoidance:
Team members must keep all files organized and double-check before deleting work.

2. Contingence:
In case of loss of work, the team members must meet up, and the person with the most dated version of the project must provide the rest with all the content and then discuss what must be redone. If a significant portion of the work is lost, the liaison must contact the line manager.

3. Minimization:
To minimize the impact of the risk, the team members are asked to save their work frequently and upload it to GitHub.

Risk (RID8): Most of functional and non-functional requirements are not met.

1. Avoidance:
Team members can keep track of all the requirements as they go ahead with the project. One of the members can create a checklist of all the requirements that have been met.

2. Contingence:
Significant deviations from the requirements require the team members to schedule a meeting with the customer and make them aware of the significant deviations from the requirements. The customer will lay down the necessary steps that are required to solve the problem. Team members should make sure they stick to the new requirements set by the customer.

3. Minimization:
To minimize the impact of this risk, the team members must set up a weekly meeting to discuss all the requirements they were able to meet and the ones that were not possible. If any member faces difficulty in achieving a said requirement, the work can be delegated to someone else or split between a few members to ease the load.

Risk (RID9): Inconsistent costumers

1. Avoidance/Minimization
Have regular meetings with the customer; thereby, we get a clear assurance of what is to be implemented. Providing them with prototype and progress at every stage of the project. Let the customer aware of all potential obligations. In this way, the client knows what they want and what they do not.

2. Contingence:
If the customer fails to be consistent with their requirements, the team members must contact the project manager. And they must be made aware of the irreconcilable behavior of the customer.

Risk (RID10): Financial problems

1. Avoidance:
Project costing should be done realistically and cleverly. Once the budget is fixed, it should not be exceeded.

2. Contingence:
Once Financial problems assize, a contingency plan must be submitted to the management.

3. Minimization:
Members of the team must be made aware of the significant repercussions of the budget being exceeded. Only after the acknowledgement of all the team members must financial decisions be carried out.

Risk (RID11): Team members may not have sufficient knowledge to do certain aspects of the project.

1. Avoidance/Minimization:
Assign tasks to the members based on their strengths and team members who are familiar with the topics should help the rest overcome obstacles. Make sure the team members have sufficient knowledge and understanding of the concept. If they face difficulties, members can provide them with videos or links that will help them understand the concepts better.

2. Contingence:
If the team member still faces difficulties with the delegated components of the project, their parts will be taken up by members with greater understanding.

Risk (RID12): Server failure

1. Avoidance: If the server fails for any reason whatsoever, there is a high risk of losing valuable data. We can avoid that by using incremental data backups, which work on backing up data only when data is added or modified and adds that to the previous backup.

2. Contingence: In the unfortunate case that this happens, we can use the last-updated back up we have for the project files.

3. Minimization: Using an incremental backup system will minimize the effects of any data lost due to server failure.

Risk (RID13): Ill-fitted photos posted

1. Avoidance: Constant monitoring to make sure any unsuitable photos do not exist on the feed, consequently giving out warnings and possibly temporary account limitations to users who belong to that criteria.

2. Contingence: If there is constant monitoring, ill fitted photos posted are bound to get removed eventually.

3. Minimization: Code an intelligent bot to report any suspicious photos.

Risk (RID14): The project targets not met by a certain time

1. Avoidance: We should set realistic timings as for when goals are to be met. We should observe our previous behavior to learn from any mistakes or bad decisions made, and possibly prioritize our timings better for more future productivity relative to the project.

2. Contingence: In case that happens, we need to contact our line manager and seek advice. We would also put more focus onto this project specifically and free up more time for it.

3. Minimization: Always make sure we are on track by staying in contact with our line manager.

Risk (RID15): DDOS attacks

1. Avoidance: DDOS attacks cannot be predicted, but we can reduce the chance of it happening by having someone monitor the firewall for any suspicious activity.

2. Contingence: In the unfortunate event of a DDOS attack, we should first contact our line manager and standby until we get proper instructions.

3. Minimization: This could be minimized by having someone constantly monitor the firewall to make sure there is not any suspicious activity in incoming traffic.

Risk (RID16): Team lacks motivation

1. Avoidance:
The team members should always motivate each other. The jobs can be allocated long in advance to keep the participants from being burnt out.

2. Contingence:
Members of the team lack motivation or feel overworked; they should approach and share their thoughts with the line manager.

3. Minimization:
To minimize the impact of this risk, team members must maintain a healthy work environment and continuously motivate, encourage, and help each other.

Risk (RID17): Illegal documentation

1. Avoidance:
Team members should access only legal documents.

2. Contingence:
Immediate meeting with the project manager and make them conscious of the illicit paperwork.

3. Minimization:
Stray free of all unlawful records.

Risk (RID18): User experience is poor, and the User interface is not aesthetically pleasing.

1. Avoidance:
Assign UI/UX to the designers in the group. Polls may be put up to see which interface Staff prefer more.

2. Contingence:
Schedule a meeting with the customer and inquire about the basic design requirements that must be fulfilled.

3. Minimization: To minimize the impact of this risk, proper research must be done before implementing a particular design. Design specifications must be based on the needs of the customer and what pleases the common eye. The latest design trends may be implemented for a better user experience.

Risk (RID19): Unable to raise extra capital if cost exceeds budget.

1. Avoidance:
Team members should create a financial structure beforehand including the project specification cost that are software costs, third party communication costs, etc.

2. Contingence:
The group members must try to raise money for the project via various sources such as private/government banks, individual asset lenders.

3. Minimization:
Regular costing should be done, that is every week/month. A buffer amount should be kept on hold for these kinds of situations.

Risk (RID20): Copyright/Patent issues.

1. Avoidance:
Use caution if it is like an already existing project. Look for the license or permissions before you use anything that is not yours.

2. Contingence:
A court case can be done which includes identifying the owner, identify the rights needed, contact the owner, and negotiate whether payment is required and hence getting your agreement in writing.

3. Minimization:
To reduce issues like this, the team should be in talks with the owner of the rights, if it is not acquired by the team. They should try to gain rights to it in the initial stages to avoid the project from getting dismissed in court.

Risk (RID20): Team member's inability to understand a spoken language.

1. Avoidance:
At the start of employing a team, the members should be questioned about their preferred/known languages. It is better for the team's communication if all team members can understand each other verbally.

2. Contingence:
The group members can easily communicate with each other in native English language using third party apps for translation verbally and text.

3. Minimization:
This can be minimized using external software or by communicating with a fellow team member who speaks a common language and can act as a translator when needed.

Risk (RID21): Team members can have problems with their workload

1. Avoidance:
Group members should let others know if they are having problems with their workloads as this can cause failure to produce work.

2. Contingence:
Work needs to be split amongst members equally for productivity. If one department is being overwhelmed with work, the team should hire extra members for that department to reduce workload individually.

3. Minimization:
Group members should not work on any task on their own, it should always be done in pairs, in this way even if one person doesn't approach the group leader, his task partner may raise this topic.

2.4. Risk monitoring

2.4.1 Background

Risk monitoring is a process where each risk is assessed and its track is monitored, i.e., whether the risk probability has increased or decreased, or whether the impact has a change. Risk monitoring is a process which goes on until the end of the project.

2.4.2 Documentation

🟡
- **Risk (RID1): Lack of group arrangement**
  Team should always be wary of their state of organization. Whenever anyone observes any unorderly behavior, a meeting should be conducted to discuss it to avoid it.

🟡
- **Risk (RID2): Code unclear to some**
  Team should be careful enough to note down any necessary explanations for any code written. In case code is still not understood by any team member(s), they should immediately contact the team and ask for a more practical explanation.

🟡
- **Risk (RID3): Poor communication**
  Roles assigned to a member or two of the group to monitor and observe the communicated behavior concurrent between the group's members. In case communication behavior seems insufficient, they should conduct a meeting to discuss it.

🟡
- **Risk (RID4): Unexpected outcomes and errors**
  Team members should always have their precautions during every stage of this project. This is to minimize the effects of any potential negative outcomes and errors.

🟡
- **Risk (RID5): Software limitations**
  The group should prioritize this matter as it is very relevant to the project. Software options and choices should be discussed in detail to avoid any irregularities midst implementation.

🟢
- **Risk (RID6): Lack of accessibility**
  The group should make sure everything they are posting and/or working with gets uploaded for the other members to be able to access. GitHub is the current solution for that.

- 🟢 **Risk (RID7): Loss of content/information**

  The team must always look at what is being added and what is being deleted. Whatever work is going to be deleted must be double-checked. After the work is complete people can push their work into their GitHub repository. The team leader can have a final look before they merge it.

- 🔴 **Risk (RID8): Most of the functional and non-functional requirements are not met**

  The team keeps track of all the functional and non-functional requirements that are met as they go forward with their project.

- 🔴 **Risk (RID9): Inconsistent customers**

  Regular meetings with the customers. We must pay attention to the complaints and demands of the customer. Keep the customer aware of all the possible work that will be done. In that way, the customer will tell us what they like and dislike beforehand.

- 🟡 **Risk (RID10): Financial problems**

  The team must ensure the financial expenditures does not exceed the set budget. Any financial commitments that are going to be made must be negotiated with the line manager and other team members; the payment must be carried out only after the approval of the line manager.

- 🔴 **Risk (RID11): Team members may not have sufficient knowledge to do certain aspects of the project.**

  The team must be made aware of people's strength and capabilities. Any member having a hard time understanding or completing given work must as for assistance.

- 🔴 **Risk (RID12): Server Failure**

  Implementation of incremental data backup as soon as possible is recommended to minimize any negative effects a server failure can have on the progress of this project.

- 🔴 **Risk (RID13): Ill-fitted photos posted**

  Setting relevant role(s) for team members should be enough to eliminate any photos in the wrong category, if any.

- 🟢 **Risk (RID14): The project targets not met by a certain time**

  Keeping contact with our line manager should be a set priority. This will help us know our position and place. This will also let us know if we need to make changes before it is too late.

- 🔴 **Risk (RID15): DDOS attacks**

  Set a role to a group member to keep watch on the firewall, in case anything looks slightly suspicious, it should be instantly reported to both the group members and the line manager.

- 🟢 **Risk (RID16): Team lacks motivation**

  Members can look out for the general demeanor of its members. Make sure everyone and everything is moving smoothly.

- 🔴 **Risk (RID17): Illegal documentation**

  Always make sure the documents that are used by the members are not illegitimate.

- 🟢 **Risk (RID18): User experience is poor, and the User interface is not aesthetically pleasing**

  The members in charge of the design aspect must take get an opinion from the people to check if the user experience is good and accept constructive criticism.

- 🔴 **Risk (RID19): Unable to raise extra capital if cost exceeds budget.**

  The team members responsible for the financials of the project should keep a regular check on the projects financial records and statements and make changes to the financial structure if costs change.

- 🟢 **Risk (RID20): Copyright/Patent Issues**

  The team members should be cautious of not using anything like an existing project or an idea. If necessary, they should seek for a license or a permission to use anything that is not yours.

- 🟢 **Risk (RID21): Team Member's inability to understand a spoken language**

  The team members should keep in touch with their colleagues often to ask if they can understand the plan of the project and if they are comfortable in the working environment.

- 🟡 **Risk (RID22): Team Members can have problems with their workload**

  The team leaders should not overload their team members. They should always be mutual when assigning over time work, i.e., not forcing it on the members. Team members should be in regular touch with each other on how they are performing and if they are comfortable in the working environment. The team members should always approach their leader if they think they are being overworked.

2.5. Definitions

2.5.1 Risk Probability

Probability	Matrix Value	Rate of Probability	Description
Low	0.25	<25%	More likely not to occur
Moderate	0.50	25-50%	May or may not occur
High	0.75	>50%	Expected to occur

2.5.2 Risk Impact

Impact	Matrix Value	Description
Low	0.25	Very little impact on project
Tolerable	0.50	Manageable impact on project
Serious	0.75	Severe impact on project

2.5.3 Risk Probability and Impact Matrix

Probability			
High	0.1875 🟡	0.375 🟡	0.5625 🔴
Moderate	0.125 🟢	0.25 🟡	0.375 🟡
Low	0.0625 🟢	0.125 🟢	0.1875 🟢
	Low	Tolerable	Serious
	Impact		

Risk Priority	Color
High	🔴
Moderate	🟡
Low	🟢

3. PROJECT DECISION AND PLAN

3.1 Project Decisions

3.1.1 Choice of Framework

We decided to go with an existing development kit for the project, it ensures we have access to documentation as we develop the system, and less time will be wasted testing code.

The flutter development kit was decided on due to its ability to support all platforms from just one codebase as well as its rich feature set. It's a new developing tool for us but we are more than prepared to learn. Flutter uses dart which is very similar to java syntax which makes it easier to pick up since we have worked on various java related projects. To handle the backend, we would use firebase which provides us with the systems database as well as a hosting service.

3.1.2 Multi-Platform Application

The system is developed to be a multi-platform application (Browser, Windows, IOS, Android). This is based on the specification provided to us. The user interface will primarily be designed using flutter and backend handled by firebase.

3.1.3 Version Control

GitHub will be used to control and manage updates to the system code as well as all documentation. This will avoid files getting over written and maintain backup versions whenever needed.

3.1.4 Communication Tools

To maintain communication with the team, we will be using Microsoft Teams as a conferencing tool for us to discuss the project.

3.1.5 Intended Audience

Based on the specification, the system needs to be targeted towards people of all ages and cultures, that are passionate about site seeing and animals. The core functionality of the application is for users to build profiles, sharing their photographs amongst a community with common interests. It will also be a place for educational outreach and organizing public meetups, where people can learn more about the area, they live in.

With that, the design of the system must be appealing towards the younger demographic as well as maintain professional designs that appeal an adult userbase.

3.1.6 Risk Management

Possible risks that could arise during the project have been addressed by the team members. A proper risk management has been documented in order provide context on what they are, and what avoidance strategies we have in place.

3.1.7 Project Costing

This is mainly theoretical, since we don't have any funding at our disposal, so all software we use is open source. We will provide a project costing file which document assumptions for each member's average wage and will assume working hours on a weekly basis. Also included would be assumed costs for the devices needed, for the development of the system.

3.1.8 Mockups

Prototypes for the system will be create using PowerPoint which will outline the layout of the system. This will then be used to collected subjective data on our design.

3.1.9 Usability Testing

Five Group members will perform usability tests, where we will use the results to come to conclusions that will help us improve the system during development.

3.1.10 Team Roles / Resources

Each member of the team will contribute to areas of the project bases on the skills and experience they have. We carried out sessions discussing our different capabilities, resulting in roles that each member is responsible for over the duration of the project.

Name	Role
Ismail El Bolock	Organizational Manager / Developer
Vishan De Silva	Technical Manager / Developer
Keval Dave	Liaison / Developer
Abdul Latif Kaissi	Reporter / Developer
Evelyn Elza	Developer
Alex Duddy	Developer
Mohammed Zain Khan	Developer
Aenon Lenvin	Developer

3.2 Project Plan

3.2.1 Overview

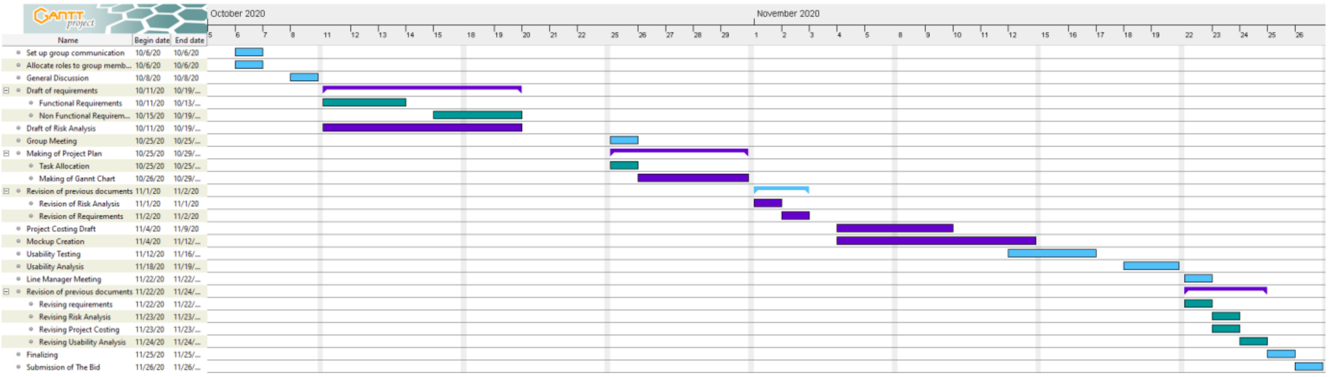
The stage 1 plan goes through each one of the tasks we wanted to achieve in stage 1 in detail. A Gantt chart has been made detailing all of the tasks and its relevant subtasks. There is an in-depth overview of which tasks must be done and who these tasks are assigned to. Stage 1 is centered around the beginning phase of our product, where we come up with new ideas and create the relevant documentation for our product.

Stage 2 and 3 are heavily focused on the creation of the product. The plan is a general overview of the tasks which we believe we will have to do for the upcoming stages. The Gannt chart here tries to replicate a agile approach to software development.

3.2.2 Stage 1 Plan

Task	Start Date	End Date	Resource
Set up group communication	10/6/2020	10/6/2020	Ismail
Allocate roles to group members	10/6/2020	10/6/2020	All
Draft of Functional Requirements	10/11/2020	10/13/2020	Ismail, Vishan, Alex, Aenon
Draft of Non-Functional Requirements	10/15/2020	10/19/2020	Vishan
Draft of Risk Analysis	10/11/2020	10/19/2020	Keval, Evelyn, Abdul Latif
Group Meeting	10/25/2020	10/25/2020	All
Task allocation	10/25/2020	10/25/2020	All
Making of Gantt Chart	10/26/2020	10/29/2020	Ismail
Revision of Requirements	11/1/2020	11/1/2020	All
Revision of risk analysis	11/2/2020	11/2/2020	All
Project Costing draft	11/4/2020	11/9/2020	Zain
Mockup Creation	11/4/2020	11/12/2020	Aenon, Vishan
Usability Testing	11/12/20	11/16/2020	Evelyn, Vishan
Usability analysis	11/18/20	11/19/20	Evelyn, Vishan

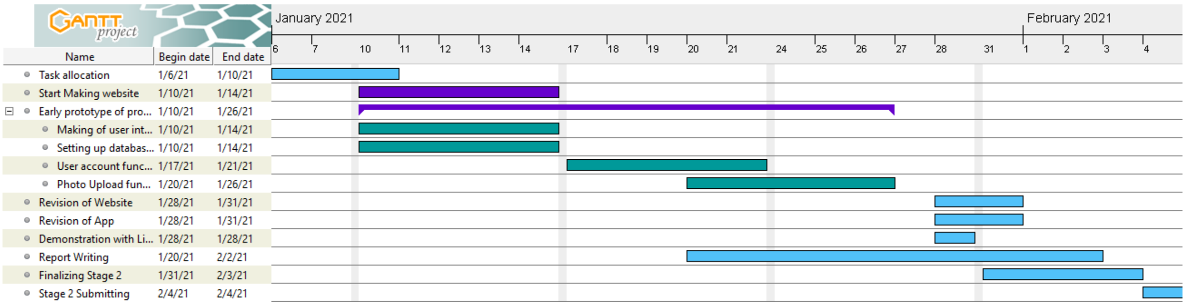
Line Manager Meeting	11/22/2020	11/22/2020	All
Revising Requirements	11/22/2020	11/22/2020	All
Revising Risk Analysis	11/23/2020	11/23/2020	All
Revising Project Costing	11/23/2020	11/23/2020	All
Revising Usability Analysis	11/23/2020	11/23/2020	All
Finalizing	11/25/2020	11/25/2020	All
Submission of Stage 1 The Bid	11/26/2020	11/26/2020	All



Name	Begin date	End date
Set up group communication	10/6/20	10/6/20
Allocate roles to group memb...	10/6/20	10/6/20
General Discussion	10/8/20	10/8/20
Draft of requirements	10/11/20	10/19/...
Functional Requirements	10/11/20	10/13/...
Non Functional Requirem...	10/15/20	10/19/...
Draft of Risk Analysis	10/11/20	10/19/...
Group Meeting	10/25/20	10/25/...
Making of Project Plan	10/25/20	10/29/...
Task Allocation	10/25/20	10/25/...
Making of Gannt Chart	10/26/20	10/29/...
Revision of previous documents	11/1/20	11/2/20
Revision of Risk Analysis	11/1/20	11/1/20
Revision of Requirements	11/2/20	11/2/20
Project Costing Draft	11/4/20	11/9/20
Mockup Creation	11/4/20	11/12/...
Usability Testing	11/12/20	11/16/...
Usability Analysis	11/18/20	11/19/...
Line Manager Meeting	11/22/20	11/22/...
Revision of previous documents	11/22/20	11/24/...
Revising requirements	11/22/20	11/22/...
Revising Risk Analysis	11/23/20	11/23/...
Revising Project Costing	11/23/20	11/23/...
Revising Usability Analysis	11/24/20	11/24/...
Finalizing	11/25/20	11/25/...
Submission of The Bid	11/26/20	11/26/...

3.2.3 Stage 2 Plan

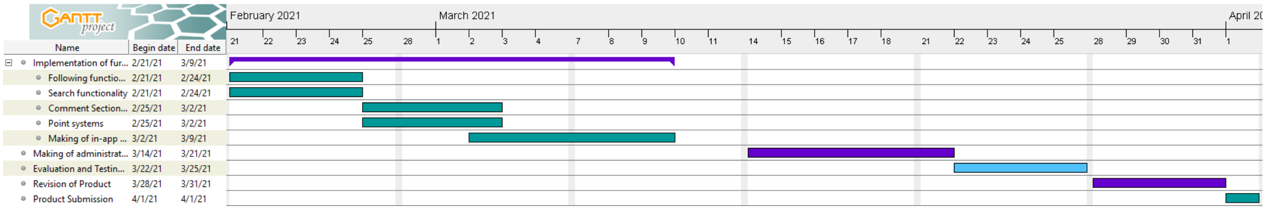
Task	Start Date	End Date	Resource
Task Allocation	1/6/2021	1/10/2021	Ismail
Making of website	1/10/2021	1/14/2021	Aenon, Alex, Zain, Evelyn
Making of user interface	1/10/2021	1/14/2021	Abdul Latif, Vishan, Keval, Evelyn
Setting up database	1/10/2021	1/14/2021	Vishan, Zain
User account making functionality	1/17/2021	1/21/2021	Ismail, Zain
Photo Upload Function	1/20/2021	1/26/2021	Abdul Latif, Alex, Keval
Revision of Website	1/28/2021	1/31/2021	Keval, Ismail, Vishan, Zain
Revision of app	1/28/2021	1/31/2021	Aenon, Evelyn, Zain
Line Manager Demonstration	1/28/2021	1/28/2021	All
Report Writing	1/20/2021	2/2/2021	All
Finalizing Stage 2	1/31/2021	2/3/2021	All
Submitting Stage 2	2/4/2021	2/4/2021	All



Name	Begin date	End date
Task allocation	1/6/21	1/10/21
Start Making website	1/10/21	1/14/21
Early prototype of pro...	1/10/21	1/26/21
Making of user int...	1/10/21	1/14/21
Setting up databas...	1/10/21	1/14/21
User account func...	1/17/21	1/21/21
Photo Upload fun...	1/20/21	1/26/21
Revision of Website	1/28/21	1/31/21
Revision of App	1/28/21	1/31/21
Demonstration with Li...	1/28/21	1/28/21
Report Writing	1/20/21	2/2/21
Finalizing Stage 2	1/31/21	2/3/21
Stage 2 Submitting	2/4/21	2/4/21

3.2.4 Stage 3 Plan

Task	Start Date	End Date	Resource
Implementing users following users	2/21/2021	2/24/2021	Vishan, Zain
Implementing search bar	2/21/2021	2/24/2021	Keval, Evelyn, Zain
Implementing comments section	2/25/2021	3/2/2021	Vishan, Evelyn, Alex
Implementing Point system	2/25/2021	3/2/2021	Aenon, Zain, Keval
Implementing Contests	3/2/2021	3/9/2021	Ismail, Alex, Zain
Implementing administration tools	3/14/2021	3/21/2021	Vishan, Abdul Latif
Evaluation and Testing	3/22/2021	3/25/2021	All
Revision of product	3/28/2021	3/31/2021	All
Product Submission	4/1/2021	4/1/2021	All



Name	Begin date	End date
Implementation of fur...	2/21/21	3/9/21
Following functio...	2/21/21	2/24/21
Search functionality	2/21/21	2/24/21
Comment Section...	2/25/21	3/2/21
Point systems	2/25/21	3/2/21
Making of in-app ...	3/2/21	3/9/21
Making of administrat...	3/14/21	3/21/21
Evaluation and Testin...	3/22/21	3/25/21
Revision of Product	3/28/21	3/31/21
Product Submission	4/1/21	4/1/21

4. PROJECT COSTING

4.1. Introduction

4.1.1 Purpose

The purpose of this entire document is to list the costing of the entire Photo-Based Social Media Platform. The document contains the entire expenditure of the entire project and individual costs as well.

4.1.2 Scope

The scope of this project is to develop a photo-based social media platform.

The main goal of this application is to provide a safe platform for the people to share the pictures that they take of objects/animals and get feedback from others and gain points as well.

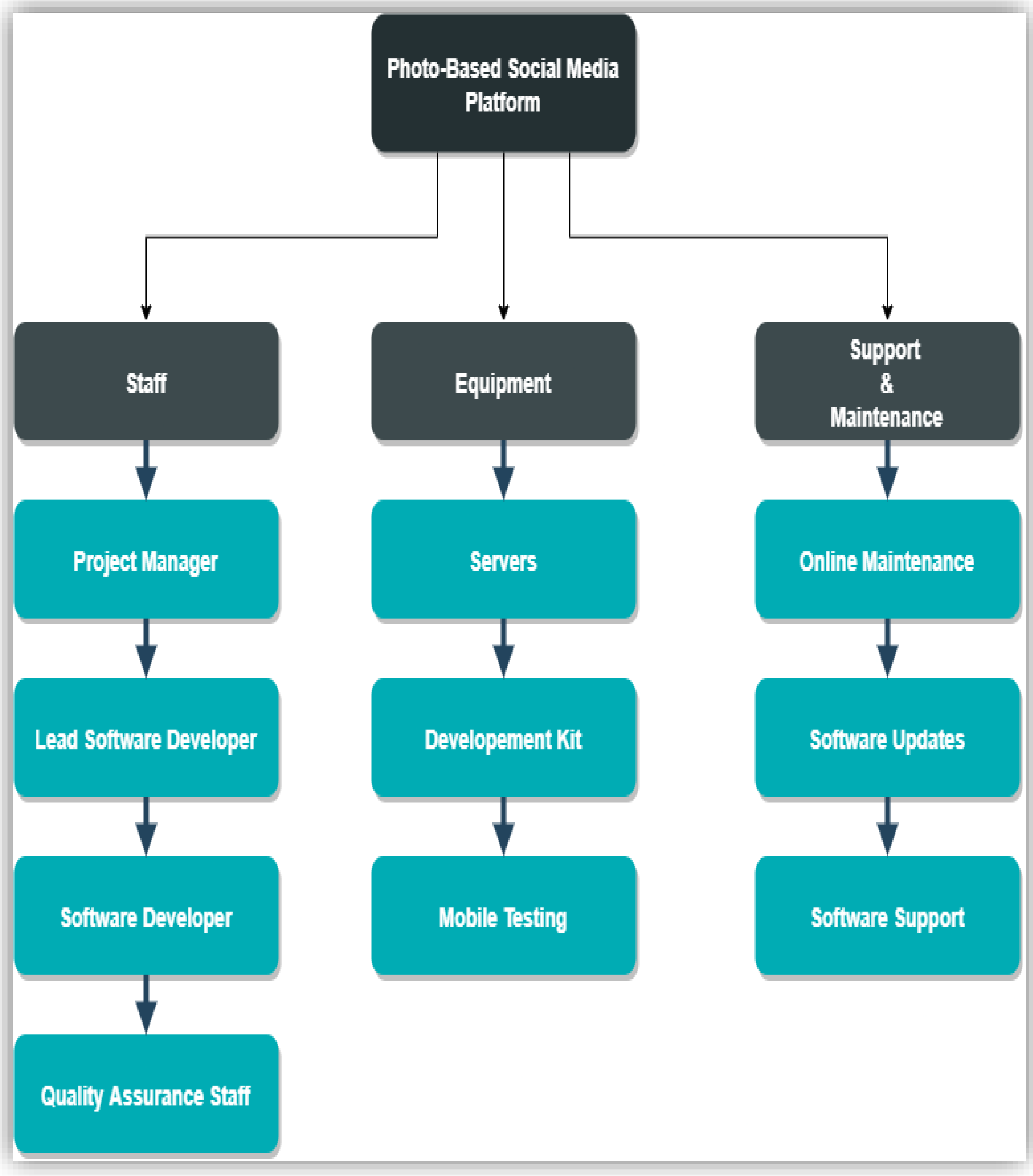
The platform even allows the people to have geolocation tagging on their pictures. This allows others on the platform to visit the places and click pictures of the objects or the animals on their own.

4.1.3 Overview

The document is to outline and summarize the list of areas where expenditure might occur. There will be a detailed breakdown of the cost provided in the document below. At the end of the document there will be a complete summarized table of the cost that will be provided to the client.

4.2. Costing Tree

Below is the breakdown tree of all the costs that will be detailed later in the final summarized table.



4.3. Cost Analysis

This section provides the written descriptions of individual cost that is mentioned in the breakdown tree above.

4.3.1 General Descriptions

Staff

Salary of all the staff members who will be involved in this project.

Project Manager

They are supposed to oversee the entire project and micro manage everything.

Lead Software Developer

They are the same as a software developer, but they lead the entire software development team and review the code.

Software Developer

They are supposed to develop the application itself; they are also possibly the ones who will maintain the application in the future.

Quality Assurance Staff

They are the ones who will inspect, test, and provide feedback to help with the development.

4.3.2 Equipment

Servers

Costs of servers that will be used during the development process, these are required for the hosting.

Development Kit

Development Kit contains the development machines that will be used to develop and run the software. These are charged at 50% of a unit cost per year.

Mobile Testing

As the application is meant to run on mobile devices, a handset (android and iOS) is required for testing the application on the device. The development machines are used for desktop testing for cost saving purposes.

Support & Maintenance

Costs of the support and the maintenance that will be required after the development process.

Support

This will be the cost of keeping the system constantly updated and fixing errors for a period of 12 months.

It includes as mentioned earlier:

- Online Maintenance
- Software Updates
- Software Support

4.4. Costing

All the costs mentioned above are now calculated and shown in the tables below. Please note in case of conflict regarding extensions or other issues, the costs are subject to a 10% charge.

Staff Cost	Unit/Hours	No. of Staff	Cost Per Unit	Wage per Person	Total Cost
Project Manger	150	1	62.16	9324	9324
Lead Software Developer	150	1	46.11	6916.5	6916.5
Software Developer	150	4	42.00	6300	25200
Quality Assurance Staff	50	2	7.25	362.5	725
				Total Cost	42165.5

Equipment	Units/Hours	Cost Per Unit	Type	Total Cost
Server	1	2000	Rented	2000
Development Kit	6	399	Rented	2394
Mobile Testing	2	500	Rented	1000
			Total Cost	5394

Support	Length	Cost
Updates and Fixes	12 months	Free*

Total Cost of System	47559.5
Included in the total cost	

Costing Refs

Project Manager Cost: https://www.indeed.com/career/project-manager/salaries

Software Developer Cost: https://www.indeed.com/career/software-engineer/salaries

General Wages: https://www.dol.gov/general/topic/wages

Development Kit

https://www.amazon.co.uk/Optiplex-i7-2600-Windows-Desktop-Computer/dp/B07YST68J3/ref=sr_1_2?dchild=1&keywords=desktop+i7&qid=1605036978&sr=8-2

5. USABILITY REPORT

5.1 Introduction

5.1.1 Aims and Objectives

This document contains a detailed plan for conducting the usability testing for our system as well as an analysis on the data we collected from our testers.

The purpose of this report is to highlight flaws and errors in the system by undergoing end-user testing and surveying. This is done to order to provide a better product that our users would be satisfied with.

5.1.2 Scope

There is one app – Novus (Photo-based social media platform)

The main goal of the application is to provide a safe platform for people to share pictures of objects/animals that they have clicked, get feedback from others on the platform and participate in photo contests.

The system will be a multiplatform application that is interacted with. The system will be developed with compatibility over multiple platforms. Which includes the Browser, Windows, iOS and Android.

Users will be able to create their own unique accounts which will be used to identify them on the system. They will be able to manage their photos shared and use the system's features to maximize their reach to their audience.

Being a photo application, the system is focused on visual content which makes it a lot easier to understand and create an enjoyable user experience. Furthermore, with the twist of a competitive environment due to the addition of contests. The userbase would have a motive to be a regular on the platform in order to perform well in said contests.

5.1.3 Overview

Test Plan

A detailed overview of how the platform will be tested. The testing methods used, the type of data we would preferably need and from whom we collect said data, what we can learn from the data we collect and how the results of the study can help us develop a product that is more suited to what our end-users expect.

Test Protocol

These are the order of tests that we would undergo in our usability study asking the user to get hands on with the program from a real-world point of view. The success rate of the users will provide us with contextual evidence that will guide the improvements to the user experience prior to implementation.

Usability Test Results

All the feedback we receive from our usability study, will be presented to summarize the general opinions from our testers about how usable the system is. Detailed demographics and stats will be provided where possible.

Conclusions

Ending this report, will be key takeaways from the usability study and any of the changes that will be applied to the system as a result.

Appendix

This will hold a downloadable link to the full usability study form we used. The formatting when trying to add it to a word document ruined its format. This study contains the consent form, test protocol questions and the post study questionnaire.

5.2. Test Plan

5.2.1 Objectives

The primary focus of the usability test is to analyze the ability of the users using our mock-up prototype, with a focus on their ability to perform given tasks regarding the functionality, with a subject group with varying levels of computer literacy. We aim to receive valid feedback that will help in identifying aspects of the system that need improving, which will be considered for implementation. To ensure varying information, testers will be recruited from a variety of backgrounds and disciplines.

The aims of this study are:

- Gather data via various forms of user-based feedback.
- Find flaws or errors with the usability application.
- Let an individual who forms a part of our target market test run the application.

5.2.2 Participants

The ideal subject group would be around 25-30 individuals who are:

- Photography enthusiasts who are looking to share their work with a bigger audience.
- Hobbyists who like taking pictures and sharing it with the world.
- Wildlife Photographers who are willing to share their professional work with a bigger audience.
- Professional Photographers who are looking to share their work.

The testers will be requested to perform series of tasks as easily as possible on the application and answer some questions about their experience over the duration of our study. In the end, they will be asked to complete a questionnaire giving their overall opinion over different aspects. Our testers are fully responsible on providing honest feedback about the platform as a whole and then on the accessibility and usability of it as well.

5.2.3 Test Scenarios

Question	Requirement
1. Feed Screen	F-UR1-5 - Photo Search F-UR1-6 - User Following F-UR2-1 - Leaderboard of Photos
2. Users Post	F-UR1-1 - Comment Section
3. Commenting on a Post	F-UR1-2 - Comment Manipulation
4. Contest Screen	F-UR4-3 - Join contest
5. Viewing a Contest	F-UR1-3 - Point Section
6. Posting Photo	F-UR3-1 - Photo Upload F-UR3-3 - Photo Editing F-UR3-4 - In-app Camera F-UR3-5 - Location Tagging
7. Profile View	F-UR8-1 - Profile Details F-UR8-3 - Follower/Following Count

	F-UR8-4 - User Posts F-UR8-5 - Profile Searching
8. Leaderboard	F-UR5-1 - Data display F-UR5-2 - Trending F-UR5-3 - Player username

Since we won't be able to supervise testers while they engage with the mock application. We have questions for each test scenario, where they can choose an option on a Likert scale that best fits their experience. Values from 1 -5, with choosing a higher value meaning the experience was better:

- 5 - Able to perform given instructions comfortably with no issues.
- 4 - Able to perform actions with slight doubt.
- 3 - Could eventually complete the task, but took their time understanding the required actions to take.
- 2 – Could partially complete the task at hand. Maybe was able to finish the task but required a lot of time to complete.
- 1 – Tester was unable to complete the task given.

These categorization's above to evaluate our testers experience, will allow us to observe the performance of our mock application. if a task has a lot of users picking values lower that three, it's a sign that there are design issues and changes need to be implemented in order to improve the user experience, for a wider range of people. It also allows us to establish an overall idea of the platform's usability.

Due to testing having to be strictly online, we won't be able to supervise our testers. Thus, they are going to be fully responsible for giving us truthful feedback about their experience with our provided tasks. From the results we received, we can confidently say that our testers did their tasks to the best of their ability and provided accurate feedback.

Tasks will be considered "complete" or "successful" when all requirements for a given task are complete. Otherwise, they are "unsuccessful" when users have either taken too long or were unable to understand our design and layout.

Specific feedback will be collected using a questionnaire that can be done at the end of the usability test. These questions are statements about their experience and testers can pick the option that best fits the given statement. Options scaling from agreeing to disagreeing with the given statements.

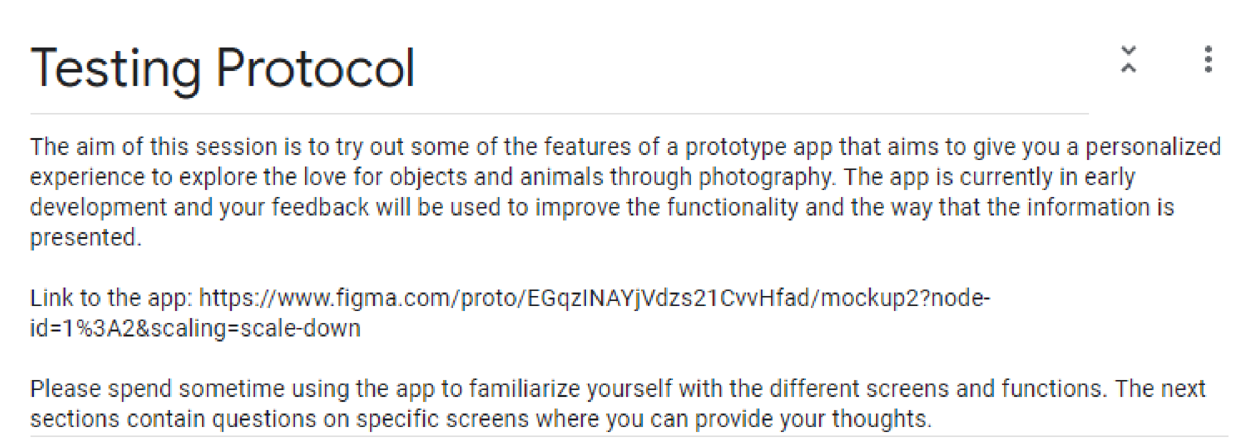
5.2.4 Metrics

Here are the main metrics that we will be using to measure and examine our subjects, participating in our study.

- Rating from 1 – 5 (Likert scale) - The higher the better, used to give a rating for our design decisions.
- Short answers - qualitative data used to understand what subjects think about our design as well as give us their views on how we can improve.
- Option rating - given some statements by us, subjects can pick the option they feel is correct about our system.

5.3. Test Protocol

For our testing methods, we decided to add specific screens to an online questionnaire along with instructions on what we would like the tester to do. This meant we wouldn't have a supervisor with the tester while he/she used the mock application. We ensured that they provide their honest experience to each given task, answering all questions truthfully. Here is our testing protocol description which is included in the form.



Testing Protocol

The aim of this session is to try out some of the features of a prototype app that aims to give you a personalized experience to explore the love for objects and animals through photography. The app is currently in early development and your feedback will be used to improve the functionality and the way that the information is presented.

Link to the app: https://www.figma.com/proto/EGqzINAYjVdzs21CvvHfad/mockup2?node-id=1%3A2&scaling=scale-down

Please spend sometime using the app to familiarize yourself with the different screens and functions. The next sections contain questions on specific screens where you can provide your thoughts.

To view the questions, there is link to where the form can download as a PDF, in the appendix section of this report.

5.4 Usability Test Results

5.4.1 Test Subject Demographic

Gender	
Male	15
Female	13
Other	0

Age	
18 – 20	23
21 – 25	5
26 – 30	0
31 – 35	0
36 – 40	0
41 – 49	0
50+	0

How often do you use social media	
All day	12
Often	12
Occasionally	2
Moderate	1
Rarely	1

What social media applications do you use	
Instagram	24
Snapchat	18
WhatsApp	21
Facebook	4
Tumblr	2
Twitter	16
Discord	16
iMessage	9

Why do you use social media	
Interacting with people	26
Staying relevant with news and events	20
Self-branding	3
Spending free time	23
Networking with people	13
Share photos/videos of my life	25
Source of entertainment	19
Share my opinion	12

5.4.2 Usability Results

Please refer to the Appendix, which has a link to the usability form used for our study. The consent form, questions as part of the testing protocol as well as the post study questionnaire.

5.4.2.1 Feed Screen

Most users were able to describe the screen they saw. There were common opinions from our testers that the application takes a lot of inspiration from Instagram and Facebook. Thus, they were able to identify sections of the screen comfortably, with some comments being “trending posts” and “posts feed”.

One user commented on the simplicity of the screen by saying:

> *“Simple and looks like a draft plan. Blue gradient backdrop with 4 trending images.”*

Task: identify all icons on screen

When asked how well they could navigate using the functions available, majority of our testers were able to identify the icons present. Here are the results on whether it was a good experience, using a Likert scale from 1 to 5.

1 (strongly disagree)	2 (disagree)	3 (neutral)	4 (agree)	5 (strongly agree)
0	0	3	6	17

Suggested changes on the feedback received:

1. Change the icon used for commenting on a user’s post. Since it’s a pencil icon, it would be better to use a chat box icon to show comments.
2. The leaderboard icon (icon with four squares) doesn’t resemble what it is meant to be. Use an icon with medals or an icon that has a podium on it.
3. The contest icon (icon that looks like a sheet of paper) doesn’t resemble a contest menu. Use a trophy icon instead.

Task: understand the layout of the feed screen

Majority of our testers were successfully able to understand the layout of the feed screen and could easily navigate it. Here are the results on whether it was a good experience, using a Likert scale from 1 to 5.

1 (strongly disagree)	2 (disagree)	3 (neutral)	4 (agree)	5 (strongly agree)
0	1	3	3	19

Suggested changes on the feedback received:

1. The screen is a bit too crowded. Icons and placement of the images are too big, better to reduce in order to make it easier to scroll.
2. Use a more futuristic and professional design. The aesthetic is too simple and outdated.
3. Make minor changes to the color choices and fonts. Use shades that are brighter as well as try working with more modern font options.

5.4.2.2 Individual posts on screen

Our testers were easily able describe the components of a user's post on our mockup. They understood how a user's posts would be represented on the feed screen, along with all the functionality a post can offer. Some comments being:

"image with comments and location"

"Emily has provided a caption to go along with her posted picture"

"Has a user post with a caption and the ability to like / comment"

Task: understand the layout of a user's post

Most of our testers were successfully able to use the functionality provided for our user post screen. This included the ability to view the account that posted it and liking/commenting the post. Here are the results on whether it was a good experience, using a Likert scale from 1 to 5.

1 (strongly disagree)	2 (disagree)	3 (neutral)	4 (agree)	5 (strongly agree)
0	1	4	8	15

Suggested changes on the feedback received:

1. Instead of having the location on the bottom, it would be better to have it on top of the screen underneath the username.
2. Have profile pictures available for users. It will help in identifying users much easier as well as allow users to personalize their accounts more.

5.4.2.3 Comments Section

When asked to describe what our testers saw, majority were able to identify it as a comments section. They noticed other comments from other users and that they could give their thoughts for the post as well. A comment from our testers being:

"Comments section for a photo with the option to give a like rating"

Task: understand the layout of the comment's screen

When asked to use the functionality of our comments section. Majority of our testers were successfully able to add a comment and give a like to the post. Here are the results on whether it was a good experience, using a Likert scale from 1 to 5.

1 (strongly disagree)	2 (disagree)	3 (neutral)	4 (agree)	5 (strongly agree)
0	1	3	8	16

Suggested changes on the feedback received:

1. Reduce the space the comments section takes. Try to make it expand so when a user clicks on it. All comments on a post become visible.
2. The comments look like a chat window. Instead of separating each one using a bubble, try to have them together in one and at the same time. Maintain good spacing to help with the readability of them.

5.4.2.4 Contest Screen

When asked about the screen, most testers distinguished the screen contains contests that they can take part in. They also noticed that scores for each active contest is visible. Some comments from our testers.

"shows contests that you can do by visiting different places"

"contest entry screen with current scores at the bottom"

However, there were a few who were not sure what the screen was meant for:

"No idea what this is for"

"not quite certain"

Task: understand the layout of the contest screen

Most of our testers were comfortably able to navigate and use all available functionality on screen. Here are the results on whether it was a good experience, using a Likert scale from 1 to 5.

1 (strongly disagree)	2 (disagree)	3 (neutral)	4 (agree)	5 (strongly agree)
0	1	2	10	15

Suggested changes on the feedback received:

1. Add descriptions to each contest, this will help users understand exactly what kind of photos you would need to take in order to do well in the contest.
2. Add some pictures as an example for each available contest. This will help users get a quick understanding of an acceptable photo for the contest. Rather than them having to click each contest to see images.
3. Make the users scores for active contests look more appealing, add some icons such as a trophy or medal and play around with colors to improve the user experience.

5.4.2.5 Viewing specific contest's

When asked about what testers thought the screen was, most understood it was related to a specific contest. In this case being the beaches contest. Furthermore, they were able to notice the functionality available to them, such as leaving like's and comments on pictures in a contest window. Some comments include:

"has images related to the beaches contest"

"entries for the contest"

"Submissions from people that a person can leave feedback on"

On the other hand, there were some who assumed the page was an explore section where you can find photos related to any topic:

"explore page"

"travel explore page"

Task: understand the layout of a specific contest window

Most testers, even though some didn't quite understand what the screen was about, were still able to navigate and select different images to leave feedback on them. However, there was one user who didn't really understand the purpose of the screen. Here are the results on whether it was a good experience, using a Likert scale from 1 to 5.

1 (strongly disagree)	2 (disagree)	3 (neutral)	4 (agree)	5 (strongly agree)
1		2	7	18

Suggested changes on the feedback received:

1. Suggested going with squared edges instead of rounded edges. This was more a common preference from our testers.
2. Reduce the wasted space on top. The title on top for a specific contest takes a lot more room than it should, would be better to work on a different layout for a specific contest screen.

5.4.2.6 Posting a Photo

Our testers could confidently identify what the purpose of the screen was, which was how users could upload images to their accounts to display to others on the platform. Some comments include:

"option to upload photo, add location and post"

"camera shooter"

"a photo ready to be uploaded to your profile"

Task: understand the functionality to upload a photo successfully

The task here was to post a photo on their profile. The process is straight forward since all there is to do is add a caption and location of the photo. After which confirming the post by pressing the "Done" button. Here are the results on whether it was a good experience, using a Likert scale from 1 to 5.

1 (strongly disagree)	2 (disagree)	3 (neutral)	4 (agree)	5 (strongly agree)
		6	6	16

Suggested changes on the feedback received:

1. The circle on the photo that's going to be posted looks weird. That was meant to show the shutter button so users can press it in order to take photos within the app.
2. The upload from phone option could be on a different screen. This would be to reduce clutter on the post screen.

5.4.2.7 User Profile

Most of our testers were able to identify this section of the app as a profile screen, which consists of the username, account description, top leaderboard scores and most importantly. All the user's posts. Here are some comments by our testers:

"Profile screen"

"Profile of an individual"

"The users personal page"

Task: understand the functionality for a user's profile

There wasn't really a specific task for them to do rather, give their thoughts if the screen had the basic requirements, they would expect of a profile screen. Here are the results on whether it was a good experience, using a Likert scale from 1 to 5.

1 (strongly disagree)	2 (disagree)	3 (neutral)	4 (agree)	5 (strongly agree)
	3	6	7	12

Suggested changes on the feedback received:

1. The user bio and leaderboard stats look out of place and awkward. Better to try different layouts and designs.
2. Have separation between the images, they look too close together making it difficult to distinguish them. Add some spacing around the boarders.
3. Waste of space, the screen looks empty in the bio and leaderboard section.

5.4.2.8 Leaderboard

Finally, we asked our testers to describe the leaderboard screen. All testers were able to identify its contents which consists of a top leaderboard, which has the top scoring users on the platform through combined contests. In addition, A scoreboard having the current users scores in active contests. Some comments include:

"shows top scores and your own score that you get by completing contests"

"Leaderboard for competitions"

Task: understand the functionality on the leaderboard screen

Here, the task was to see if all our testers could easily distinguish the content on screen. That being the two different leaderboards as well as differentiating the content on each respective leaderboard. Here are the results on whether it was a good experience, using a Likert scale from 1 to 5.

1 (strongly disagree)	2 (disagree)	3 (neutral)	4 (agree)	5 (strongly agree)
	2	4	7	15

Suggested changes on the feedback received:

1. Use more icon's next to the leaderboard. Would be useful in order to identify them better as well as to spice things up.
2. This screen could be merged with the contest screen, instead of having it as a stand-alone window. Combine the contest and leaderboard and use the available slot on the navigation bar for something else.
3. Improve the design choices, the layout is fine but plain. Make it more exciting.

5.4.3 Questionnaire Results

Below is a table consisting of the number of responses for our usability questionnaire.

*colored boxes represent the highest value

Question	Strongly Agree	Agree	Possibly	Disagree	Strongly Disagree
Logging or signing up to the app is effortless	11	13	3	0	1
Icons are well placed and easy to understand	9	14	3	2	0
Design of the app is attractive and simplistic	7	5	3	12	1
Uploading photos is a straightforward process	14	9	5	0	0
Navigating the app is easy	8	12	8	0	0
The app is well structured and user friendly	8	9	10	1	0
Layout for contests can be understood	8	14	4	1	1
Layout for user leaderboards can be understood	9	14	3	1	1
Sending and reading comments is trouble-free	8	14	5	1	0
Able to search for users and taglines on the platform	10	9	7	1	1

5.5 Conclusion

Through the usability study we performed, we received valid feedback that we can use to improve our existing prototype. This study helped arise a lot of mistakes and assumptions we made towards our design and now, we have a better understanding about how the users feel about the product we are developing.

In general, the testers understood majority of the screens we asked them to play around with. Most were able to navigate with very little guidance while a few, were just not sure about the purpose of certain screens. Adding to this, there were numerous comments on our design choices and layout being too basic and dated. Other than that, the core functionality we had in place was understood by all.

5.6 Appendix

Link to Usability Test Form (as PDF file) - https://heriotwatt-my.sharepoint.com/:b:/g/personal/vsd2_hw_ac_uk/EehuCJeJ7BtBjSYIbenIDpcBh7DOjZqXRWE5FG0yWzvi2g?e=46tocl

6. DOCUMENT DETAILS AND HISTORY

6.1 Document Copyright

Document Title	Confidentiality Level/Status	Copyright
The Bid	Final Version	© 2020 VS, IE, KD, ER, AL, AK, MZ, AD

6.2 Document Version Table

Date	Version	Authors	Changes
20/10/20	Initial draft	VS, IE, KD, ER, AL, AK, MZ	N/A
09/11/20	Created functional requirements	VS, IE, AL	Capability Requirements
09/11/20	Added Risks	AK, ER, KD	Risk Analysis
10/11/20	Added Constraints	VS, IE	Constraint Requirements
11/11/20	Added Project Costing	MZ	Project Costing
11/11/20	Updates to Risk Planning	AK, ER, KD	Risk Analysis
20/11/20	Final Project Costing	MZ	Project Costing
21/11/20	Team Roles added	VS	Project Plan
21/11/20	Created Mockup and Novus logo	AL, ER	Mockup
22/11/20	Risk Monitoring	AK, ER, KD	Risk Analysis
22/11/20	Added Matrix Definition	KD	Risk Analysis
23/11/20	Added Use case folder	VS, ER, AK	Appendix
24/11/20	Usability intro and test plan	VS, AL, MZ	Usability
25/11/20	General Description	IE	GD and Reqs.
25/11/20	Final Changes to all documents	ER, AK, KD, VS, IE, AL, MZ	*All Documents*
26/11/20	Final formatting	IE, KD	Novus Stage 1

6.3 Minutes of the meeting

Meeting No.	Date	Topic	Time
1.	14/10/20	Initial Draft Points	15:00-17:00
2.	20/10/20	Implementation of Ideas	15:00-17:00
3.	3/11/20	Discussion of further docs.	15:00-17:00
4.	10/11/20	Implementation of Reqs. and Risks	15:00-17:00
5.	17/11/20	Project Costing along with further additions to Risks and Reqs.	15:00-17:00
6.	19/11/20	Meeting with Line Manager. Discussion about progress so far.	15:00-17:00
7.	22/11/20	Mockup and Logo. Further discussions on project.	11:00-12:00
8.	24/11/20	Use Case Diagrams, Usability and Gen. Desc. Added.	13:00-15:00
9.	26/11/20	Final Discussions about the stage 1 document.	12:00-13:00