Artyfy - Quality Document

Project Team 5

Advanced Software Engineering Topics

Comp8117

Dr. Aznam Yacoub

Purpose:

The goal of our project is to create a high-end, user-friendly, and cost-effective Gallery and Art Management engine that will assist the intended target users in keeping track of all area galleries, museums, artists, and art exhibitions. This programmer connects existing artistic platforms (Art Galleries & Artists) and can create a new defined environment for art enthusiasts. The Art Finder application will classify paintings based on their Art Style using picture and pattern recognition techniques. By just uploading the image of the art, students and museum personnel will be able to put the numerous artworks into the appropriate category.

Scope:

The scope of this project is to create a genuine Android application that allows artists and art gallery owners to promote and sell their works of art. The app can also be used to find nearby art galleries and museums.

Structure of the team:

All the team members who have contributed to creating Artyfy have deep knowledge about how the software engineering lifecycle model works as everyone has contributed up to the limit. Almost everyone was new to Android Development, so it took more time than expected to learn android but the practical implementation on android was far faster than expected as the things we have decided, all the team members have adapted those in a lesser amount of time. The structure of the team was as follows:

Team member name	Role
Yuva Venkata Sai Duppati	Scrum Master / Full-Stack Developer
Bhargav Sai Karicheti	Full-Stack Developer
Shruthi Paka	Full-Stack Developer
Lakshmi Vardhani Pravallika Munukutla	Full-Stack Developer
Lakshmi Narayanan Shankar	Full-Stack Developer
Nupur Badiani	Full-Stack Developer
Deepaksai Valluru	Full-Stack Developer
Varun Ahuja	Full-Stack Developer

Everyone in this team has individual tasks assigned and which were completed on time and the overall progress was on track every time as verified by the customer as well. All the Scrum-related tasks are mostly maintained by the Scrum Master who was supported by every team member as well as he worked in the core development as well.

Software Development Life Cycle

The SDLC model that is used for the development of the project is Agile. In Agile methodology, the prime objective is to satisfy the customer's need through an environment of transparency in communication, collective responsibility, and continuous progress. There are so many frameworks available to implement Agile such as Scrum, Kanban, etc., but we have used Scrum for Artyfy. Scrum is an agile methodology in which products are developed iteratively. You can focus on the product development work if you are using the right tool which has the power to manage all other tasks. There are some integral parts of scrum methodology which include Planning, sprints, standups, and retrospectives.

The processes how we implemented the SDLC model in Artyfy are described as:

- During Sprint Planning, the major tasks in the sprint are divided into sub-tasks and are distributed among team members. Each team member can track the record of their task from the scrum board or JIRA board for the overall progress of the application.

- Information about each task is shared with each team member during the daily scrum meetings.

- If anything can be picked from the backlogs, then it's added to sprint tasks.

- After the sprint gets over, a sprint review is done to track the overall progress of the product till the sprint.

- If there is any task that is left uncompleted, or any bug found during the sprint completion then those tasks are put in the backlogs.

- A sprint report is to be made for the progress.

- In the end, a sprint retrospective meeting is conducted and the report for that is created for submission.

Communication Procedures:

Working in Agile methodology, provided us a great source of interacting with other team members and to track the overall progress, i.e., Scrum Board or JIRA board on which we had all the processes that were running in the sprint as well as what else need to be done. Scrum meetings help team members to interact with the team members to work collaboratively as this is the best source of communication channel that we had to transfer information and to be aware of other team members if they are stuck in the same situation. According to Scrum, a team needs to have a daily scrum meeting to share the progress and to cope with the upcoming tasks as well as to clarify to the Scrum Master what all tasks are undergoing and what are upcoming. This acted as a communication channel or messaging queue for all the team members, and we shared the information and further, due to this, we managed to complete the overall project within the deadlines after facing so many hurdles due to some circumstances.

The Scrum Process that we followed in Artyfy consists of all the processes contained in Scrum Methodology, that are:
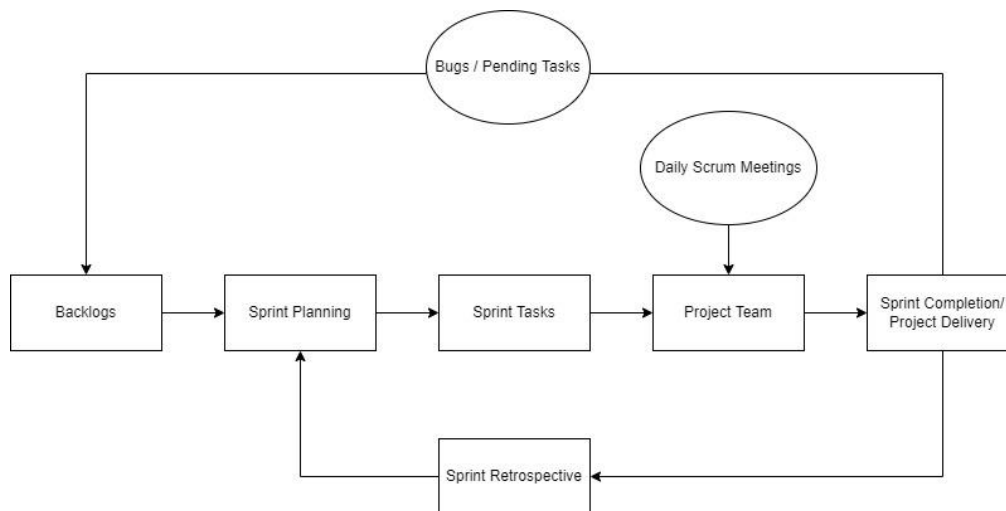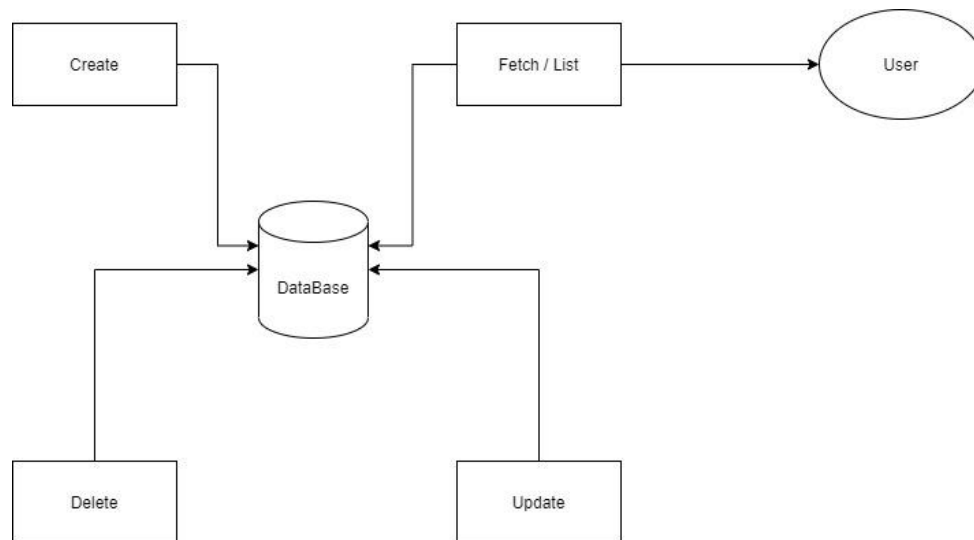

Development Procedures:

The whole development procedure was following the Agile SDLC process and according to that, the development process was completed. While learning Android Development, we were not sure if we could be able to complete the project within the deadline but according to the methodology we followed, we coped up with the progress, and with a little slip in the first sprint, we managed to complete the development process on time and within the deadline.

The development process involves the development of all the users and admins that can run the application with their front-end screens and backend connectivity to connect the application with the database so that the database is saved to a cloud-based database, i.e., Firebase. Then, the process involved the Art-Finder feature which includes finding any art with Machine Learning Algorithm. All the users created had some CRUD operations linked to the application and further to the database which suggests that the data entered for any user is first authenticated and then authorized by the application following with the application of logic and ultimately reached the database for future references. All operators of this application can have certain operations available such as Create, Read, Update and Delete, to perform the mentioned transactions.

The flow for these processes is as follows:

Versioning Procedures:

The development procedure of Artyfy consists of a step that involves deploying the code to a repository that can maintain the overall progress according to the tasks completed and picked from the Scrum Board. Git was used as a tool that has managed the overall process by providing appropriate versioning with the help of a platform that was used by the team members to upload the code, i.e., GitHub. Git has provided the flexibility that helped all of us to maintain a systematic view of the application with its versions and one can easily get to the version that has been completed in a specific sprint. GitHub has provided a platform that has helped us to save all the code with branches and to save the code with appropriate versioning.

According to the systematic project development, all the versions need to be maintained as it helps the user or developer to switch to any version that is stable or to shift to any specific version for a testing process that may involve white-box testing, black-box testing, unit testing or functional testing of the end-to-end system. Thus, Git and GitHub have helped us to maintain the overall versioning of the project and helped us to track as well as test any specific feature with a compatible version by just switching or jumping onto a specific version.

Risk Management

Risk management is the systematic process of identifying, analyzing, and responding to project hazards. It comprises increasing the chances and implications of positive events while reducing the chances and consequences of negative ones. The effect of unfavorable events on the project's objectives. A risk management strategy lays out how a business will handle its hazards. The project team will manage risks in order to achieve that goal.

Risk management comprises containing and managing risk. To get started, you'll need to figure out what you're going to do and plan. Then, when a risk arises, be ready to act, drawing on the entire team's experience and knowledge to reduce the project's impact.

Risk:

An unforeseen events or circumstances that, if they occur, have a positive or negative impact on the project's objectives. Risk is a three-part metric that assesses the inability to fulfil overall project objectives within set project requirements and constraints:

(1) The impact of the occurrence.
(2) The impact of the risk on the program
(3) The time frame in which the risk's consequences will manifest if it is not mitigated.

Chances of Occurrence

The chances of occurrence can be defined as follows:

Range Of Occurrence	Occurrence	Numeric score
91% - 99%	Very likely	5
61% - 90%	Likely	4
41% - 60%	Moderate	3
11% - 40%	Unlikely	2
1% - 10%	Very Unlikely	1

Impact Of the Risk If It Occurs

In the table below, the risk effect categories and phrases are defined. Consider how the effect description for positive risks is the opposite of the effect description for negative risks. The situations would be same, with the exception that they would be beneficial to the project.

Impact	Effect Of Impact	Impact value	Score
An occurrence that might result in considerable cost and schedule increases if it occurred. It's possible that low-priority requests will not be fulfilled.	Critical	Cost of risk	10
An occurrence that might result in considerable cost and schedule increases if it occurred. It's possible that low-priority requests will not be fulfilled.	Serious	Cost of risk	8
An event that, if it occurs, would result in mild cost/time increases while maintaining major functionality.	Moderate	Cost of risk	5
An incident that has no bearing on the functionality of the system	Negligible	Cost of risk	1

Risk Response:

For each danger that has been recognized, a response must be developed. For each risk, the project team oversees selecting a risk response. The project team will need the greatest possible risk assessment and description of the response options to determine the optimum reaction for each risk. The probability of a risk event occurring, and its repercussions will be utilized to decide the extent to which risk mitigation measures should be implemented.

Risk Score

The risk score is a calculated value that is the product of the chance of occurrence and the magnitude of the impact. You use the score to compare risks as part of the risk prioritization process. The risk score was calculated using this matrix. From 1 (very low exposure) to 50 (extremely high exposure), the values range from 1 to 50. (Very high exposure). The terms "low," "moderate," and "high" are defined as follows:

- **Low Risk:** Has a low to no chance of expense increases, schedule disruptions, or performance degradation. When they arise, the project will keep an eye out for them and deal with them.

- **Moderate Risk**: It's likely that the price will rise, the schedule will be disrupted, and the quality of the work will suffer. The project will develop a response strategy for these threats.

- **High Risk:** Likely to result in significant expense increases, schedule disruptions, or a drop in performance. The initiative will create solutions for these dangers.

Following are the risks involved

- **Privacy and Security:**

Determine where personal (phone number, email address) or sensitive (payment and password) information is needed. When storing this type of information, security testing should be a top priority to prevent leaks, hacks, and breaches from tarnishing reputation. We will ensure that the user's privacy and security is not at threat by all possible means.

- **Compatibility:**

Application should be tested on variety of browsers and devices during exploratory testing when it undergoes a code update. Application versions come in a variety of setups; therefore, you should test it on all of them. Before launching the application will ensure it is compatible with all the versions of android and tablets.

- **System Reliability:**

An application must avoid failures such as ISP server crashes, online payment system faults, and plugin bugs. All these difficulties have the potential to bring the entire application crashing down, which would have a significant impact on consumer satisfaction. The best way to overcome this problem would be to run the application in all environments and even going through the worst-case scenario and have the solution ready if it occurs.

- **Data Correctness**:

The details mentioned in the application should be correct and precise. The false data can lead to losing the trust of user. So, the information regarding the art, artist and nearby museum should be double verified. The data entered and the data would be entered by the artist would be double verified and then uploaded to maintain the data integrity.

Quality Assurance Plan

Purpose:

The purpose of the Quality assurance plan is to ensure that the product meets the objectives throughout the project lifecycle. The Quality Management process is iterative and incremental. The process encompasses quality requirements identification, planning, implementation, and execution of the plan. Once the plan is operational, the Quality Assurance(QA) team leverages the plan to assess, measure, monitor, and continually improve the plan.

Scope:

This Quality Assurance Plan provides a foundation for managing the Artyfy software quality assurance plan.

- The document will identify the SQA responsibilities of all the members of this project.
- The document will define the reviews and audits and how to conduct these.
- The document will list the activities, processes for the Software Quality

Quality Objectives:

For the Software development - process,

- We will ensure that the software will satisfy all functional and technical requirements.
- We will ensure that the software maintenance activities will take care of managerial scheduling and budgetary requirements
- We will be managing activities to increase the efficiency of the software quality assurance activities.

Quality Assurance Management

Organization:

This section describes the organization's structure required to maintain and enhance the quality of the software.

The Software Quality Assurance (SQA) Personnel should work independently. The SQA Personnel will report directly to the Project Lead rather than the team members. This independence will provide the SQA Personnel to maintain the quality of the software product and make accurate decisions. If the SQA Personnel encounters any scenario which can jeopardize the software quality, they should enlighten the team members and the Project Lead. The SQA Personnel works above the project team. It would enable the Artyfy SQA Team to resolve various problems at the project level itself.

Tasks and responsibilities

The SQA team is responsible for ensuring that the project team complies with the required Software Quality Assurance plan. They assure the quality of the software delivered and all the related documentation.

Following is the description of the tasks and responsibilities of each group to provide quality software.

- o Scrum Master is responsible for

    - Reviewing and approving the SQA plan
    - Setting a quality program by engaging the project members to implement the Quality Assurance plan.
    - Following-up on quality issues raised by SQA team
    - Identifying the quality measures to be implemented in the software
    - Identifying the activities to be performed by SQA
    - Ensuring that defined processes are being followed by respective teams

- o Artyfy QA Team is responsible for

    - Review and comment on the SQA Plan
    - Checking the quality factors are implemented in the software
    - Performing the quality program following the SQA Plan

- o Artyfy Design Team is responsible for

    - Resolving any quality issues raised by SQA Personnel related to software design implementation
    - Performing the quality program following the SQA Plan
    - Implementing the software design practices, and procedures as defined and prepare project planning documents

- o Artyfy Development Team is responsible for

    - Resolving any quality issues raised by SQA Personnel related to software development implementation
    - Performing the quality program following the SQA Plan
    - Unit and first level of Integration testing which validates that component work together.

- o Artyfy RnD Team is responsible for

    - Independent review, and validation of features such as the data model and application structure.
    - Performing the quality program following the SQA Plan.

Test Report

Project Details:

Testing refers to the process of evaluating an app. This procedure is beneficial in preventing numerous problems that may arise during the software development process, as well as adding value to the product by ensuring it fulfils all of the client's requirements.

Name	Artyfy
Description	This app offers a list of surrounding institutions as well as a total number of artists who have registered to present their artwork and sculptures to art lovers and the general public. This mobile application reduces the barriers that limit people's access to art by allowing them to enjoy the work of various artists or creative locations.
Test Objective	Examine the product's stability, as well as its functions and features. The user's judgement of product stability, which covers quality, performance, dependability, and functionality of product themes, is known as stability acceptance.

As an app is being constructed, the most important aspect is its design, which must be user-friendly. After all, application is all about the user, and if they find it difficult to use, they will not utilise it, resulting in a low conversion rate. Then, to verify that all of the functions are working properly, proper testing is required, which aids in the elimination of any bugs or faults that may arise during use. It is critical to make certain of this.

How Testing Can Be Beneficial:

Here are some key areas where application testing may be quite beneficial:

- **It verifies a number of things:** Application testing also allows you to assess the usability, functionality, performance, compatibility, and security of a website or app. These are important considerations for any app.
- **Improves usability**: The usability is critical, as is addressing the risk factors that come with it, both of which can be accomplished.
- **It ensures that transactions are completed on time**: Testing ensures that each and every page on the application is carefully tested and verified to ensure that all

transactions are completed successfully. This also aids in determining whether or not the app is ready for usage by the customer.

- **Examines functionality and compatibility in relation to load:** To ensure that client requirements are met flawlessly, we must ensure that all the functions available on tare properly functioning under a variety of load conditions. Organizations must also ensure that their app is compatible with a variety of browsers and systems. All of these concerns can be addressed through testing, which also ensures compatibility and decent performance under severe loads.
- **Reduces the risk of change:** It is also easy to ensure that the integration of new modifications does not damage the performance or produce any unintended issues by conducting testing.

Following Risks Can Be Prevented by Performing Testing?

- Slow application loading
- Weak search functionality
- Unavailability of item
- Could not sign in
- Wrong artist details
- Wrong display details

Important Test Cases:

As a result, test cases are primarily scripted/non-scripted scenarios used to verify the various functionalities of an application. And we've included a complete checklist as well as key test cases that an application testing team should focus on.

1. In General

It is critical that the general Test cases for app are carefully written and thorough. Even the tiniest details, such as the home screen interaction quality, how smooth the navigation across product categories is, or whether the product picture enlarges upon clicks, and so on, require a great deal of care. The following are some of the general tests that must be run on the website:

- Whether or not the links are routed to the right art/category pages

- Whether the art, category name, art description, and artist details are all clearly visible.
- Whether all category/museum pages include a relevant art that is specifically listed for the category/museum.

S.No	Action	Input	Expected Output	Actual Output
1.	Whether or not the links are routed to the right art/category pages	Click on any art picture	Able to direct to the clicked art details	Directed correctly
2.	Whether the art, category name, art description, and artist details are all clearly visible	Click on any art picture	All details displayed are correct and accurate	Details were correct and accurate
3.	Whether all category/museum pages include a relevant art that is specifically listed for the category/museum	Click on any particular gallery and match the art with category	Art should be displayed in its category only	It was displayed correctly

2. Art finder and art suggestions

In applications, this is the most popular option. Even with an easy and wide category design, customers may have difficulty finding the art they are looking for. This is why it's critical to test the search tool, since it makes it easier for buyers to find things quickly and easily. The following items should be evaluated in search functionality:

- Is the search available based on the painting name, artist name, or other criteria?
- Sort items based on artist, category, and other factors should be sorted well
- If the results are on numerous pages whether there are alternatives to navigate between them.

1.	Is the search available based on the painting name, artist name, or other criteria	Search for a particular art, artist, or museum	Should display the correct search or closely matched results	Displayed correctly

2.	Sort items based on artist, category, and other factors should be sorted well	Sort using any one of the options	Sorting should be correct and accurate	It was correct and accurate
3.	If the results are on numerous pages whether there are alternatives to navigate between them	Click on back button after viewing an art	Should be taken back to the previous tab	It was displayed correctly

3. View arts & painting

An art display page's testing is a critical part. Testing verifies that the arts listed are fully functional. Important information such as painting details, artist details and art specification can be found on the display page, and it is critical to ensure that all of this information is displayed correctly every time a consumer logs into the app. It is critical to test the following:

- Are the details of the art displayed?

- Are the image/images displayed correctly?

- Is there any information about museum where it can be found?

S.No	Action	Input	Expected Output	Actual Output
1.	Is the details of the art displayed	Display art details	Should display the correct details of art	Displayed correctly
2.	Is the image/images displayed correctly	Images of art or paintings	Correct images of art should be displayed	It was correct and accurate
3.	Is there any information about museum where it can be found	Click on info button to get information	Should display museum information where is it found	It was displayed correctly

4. Buy and Payment Options

- Payment is done without any problem

- In case of payment fails due to any error

S.No	Action	Input	Expected Output	Actual Output

1.	Payment is done without any problem	To check if payment is done correctly	Payment should be done easily	Worked well
2.	In case of payment fails due to any error	To check cases in case of payment failure	Should refund the entire in the account in 3-4 business days	We intentionally created an error to check the output, money was returned in 5 days

5. Hardware and software compatibility

- Measure hardware compatibility and performance utilizing required devices by running a hardware test against a variety of phone and tablet interfaces.
- Check for compatibility across all android versions.
- Check performance for all android versions

S.No	Action	Input	Expected Output	Actual Output
1.	Measure hardware compatibility and performance utilizing required devices by running a hardware test against a variety of phone and tablet interfaces	To check on variety of phone and tablet interfaces	Should work well for all devices	Worked well
2.	Check for compatibility across all android versions	To check compatibility for all android versions	Should operate well on all versions	Operated well on all versions
3.	Check performance for all android versions	To check performance on android versions	Should perform well and same for all android versions	Performed well for all versions

6. User login and registration

- Validate the registration page contains all of the desired input boxes (Username, Email Id, Password, Confirm Password, First Name, Last Name, Phone Number, Date of Birth, Gender, Location, Acceptance of Terms of Use, Acceptance of Privacy Policy, Submit, and a link to the Login page).

- Verify if a user will be able to login with a valid username and valid password.
- Verify the 'Forgot Password' functionality

S.No	Action	Input	Expected Output	Actual Output
1.	Validate the registration page contains all of the desired input boxes	To check if registration is done.	The user should be registered	Worked well
2.	Verify if a user will be able to login with a valid username and valid password.	To check if the user is able to login.	The user should be able to login successfully.	Operated well
3.	Verify the 'Forgot Password' functionality	To check if the user is able to change password incase, he doesn't remember	The user should be able to get a link to change password	Performed well

7. Database testing

- Check if the correct data is getting saved in the database upon registration and login.

- Check if the correct data is getting saved in the database upon addition, updating and deletion of art.

S.No	Action	Input	Expected Output	Actual Output
1.	Check if the correct data is getting saved in the database upon registration.	To check the database for registration and login information.	The user information and login details should be available	Worked well
2.	Check if the correct data is getting saved in the database upon addition, updating and deletion of art.	To check if the changes to image are getting saved	Images should be saved properly	Operated well on all versions

8. Deployment and release.

We have done a final test on the application before release checking for the following categories.

- Functional tests: does the app work?
- Compatibility tests: does the app work consistently for everyone?
- Performance tests: does the app respond quickly and how does traffic affect performance?
- Security tests: is the app secure against attacks?
- Usability tests: is the app easy to use and does it respond to interaction as expected?

S.No	Action	Input	Expected Output	Actual Output
1.	Functional tests	To check if the application is working well	The application should function well without and bugs.	Worked well
2.	Compatibility tests	To check compatibility for all android versions and if it is consistent from user to user	Should operate well on all versions and should be consistent with the every user	Operated well on all versions
3.	Performance tests	To check performance of the app is quick and if huge traffic makes the application slow.	The application should perform well with huge amounts of traffic as well	Performed well
4.	Security tests	To check if the application is safe from attacks.	All the user data should be safe and secure	Worked well
5.	Usability tests	If the app is user friendly.	The application should be simple and easily understood by the user	Works well.