**Project Proposal**

**Project Name: "AnyScript"**

**Problem Domain:** Category A

**Technology:** Web, Mobile Phone/Handheld Devices

**Background:** Internet is full of Text based content. Clients often struggle to find talented writers who can produce high-quality copy within tight deadlines and can meet their specific needs and requirements. Talented writers on the other hand often find difficulties finding suitable work or clients.

**Objective:** The objective of AnyScript is to provide a platform that connects clients with talented writers who can produce high-quality copy or content within tight deadlines. AnyScript will make it easy for clients to find reliable and talented writers, and will help writers build their portfolios and expand their careers.

**Target Group:** The target group for AnyScript is clients who need high-quality writing services and writers who are looking to build their portfolios and expand their careers.

**Solution:** AnyScript will provide a web-based and a mobile platform for businesses and individuals to post writing projects and connect with talented freelance writers. The platform will use a combination of matching algorithms and a rigorous screening process to match clients with the right writers for their needs. Clients will be able to review the portfolios and ratings of writers and choose the best match for their project. Writers will be able to showcase their skills and experience, bid on projects, and receive fair compensation for their work.

**Basic Functionalities:**

* A platform for clients to post writing projects and receive bids from freelance writers.
* A matching algorithm to match clients with the most suitable writers for their project.
* A portfolio and rating system for writers, allowing clients to easily evaluate their skills and experience.
* Secure payment processing for clients to pay writers for their work.
* Communication tools for clients and writers to collaborate effectively.

**Premium functionality will include:**

* Priority matching with top-rated writers.
* Access to a larger pool of potential writers.
* Enhanced project management tools.
* Customizable bids and proposals.
* Writers can pay extra to have their profiles and services appear at the top of search results, increasing their visibility to potential clients.
* Clients and Writers can receive fast, dedicated support from the platform's customer service team.

**Business Model:**

* The company earns revenue by charging fees on transactions between Writers and clients

* Writers can pay extra to have their profiles and services appear at the top of search results, increasing their visibility to potential clients.
* Subscription fee for Premium Features.
* Writers can pay extra to have their profile and services appear at the top of search results increasing their visibility to the potential client.
* Writer can create custom offer for the client, allowing them to offer specialized service at a premium price.
* Writer can show their best works in an unlimited dedicated portfolio section for premium subscription, helping them stand out from the competition.

**Functional Requirements**

**2.1** **Registration and Login**

**2.1.1** The software shall allow users to create accounts with their basic information.

**2.1.2** The software shall allow registered users to log in with their username and password.

**2.1.3** The software shall verify the login credentials with the database records.

**2.1.4** If the login is successful, the home page of the user account shall be displayed.

**2.1.5** If the login fails, a random verification code shall be generated and sent to the user’s email address to retry the login.

**2.1.6** If the number of login attempts exceeds the limit (3 times), the system shall block the user account login for a specific period (optional function).

Priority Level: High

**2.2** **Job Posting**

**2.2.1** The software shall allow clients to post jobs with the required details (title, description, deadline, budget, etc.).

**2.2.2** The software shall allow clients to choose the desired writer(s) or select the automatic matching option.

**2.2.3** The software shall allow clients to communicate with the selected writer(s) via the messaging system.

Priority Level: High

**2.3 Bidding and Accepting Jobs**

**2.3.1** The software shall allow writers to browse the available jobs and bid on the desired ones.

**2.3.2** The software shall notify the client of the received bids.

**2.3.3** The software shall allow the client to select the desired writer(s) or select the automatic matching option.

**2.3.4** The software shall notify the selected writer(s) of their acceptance and the required details (deadline, budget, etc.).

**2.3.5** The software shall allow the selected writer(s) to communicate with the client via the messaging system.

Priority Level: High

**2.4 Communication and Collaboration**

**2.4.1** The system shall provide a messaging interface for clients and writers to communicate.

**2.4.2** The system shall allow clients and writers to exchange project files and documents.

**2.4.3** The system shall provide a project management interface for clients and writers to collaborate on a project.

Priority Level: Medium

**2.5** **Payment and Rating**

**2.5.1** The software shall allow clients to make payments for the completed jobs via the integrated payment system.

**2.5.2** The software shall allow clients to rate the completed jobs and writers based on clients’ satisfaction.

**2.5.3** The software shall allow writers to withdraw their payments to their specified accounts.

**2.5.4** Anyscript will receive a 7% cut of each payment.

Priority Level: High

**2.6 Writer Homepage/Portfolio Update**

**2.6.1** Writers will be able to update their profiles

**2.6.2** Writers can upload their portfolio and show their works and experiences

Priority Level: Medium

**Non-Functional Requirements**

**3.1** **Performance**

**3.1.1** The software shall support a large number of concurrent users without performance degradation.

**3.1.2** The software shall ensure fast response times for all user interactions.

**3.1.3** The software shall ensure 99.99% uptime for the platform only exception server maintenance and that to be fixed within 2 hours.

**3.2** **Security**

**3.2.1** The software shall ensure the confidentiality and integrity of user data.

**3.2.2** The software shall ensure secure and end to end encrypted communications between the platform and the users, client and writer.

**3.2.3** The software shall ensure secure storage and retrieval of user data.

**3.2.4** The session will log out after certain period.

**3.3** **Usability**

**3.3.1** The software shall provide a user-friendly interface for all users.

**3.3.2** The software shall provide clear and concise instructions for all user interactions.

**Project Requirements**

**4.1** **Tools for Web Development**

**4.1.1** The software development team shall use industry-standard tools (VisualStudio as IDE, GitHub as Repository for version control, branching and merging) for the development process.

**4.1.2** The software shall be developed using the HTML, CSS, JS, Python, C# programming languages and frameworks(Tailwind CSS, Blazor, LightHouse, Google Dev Tools, Vite etc).

**4.1.3** The software shall be tested using automated testing tools (e.g., Selenium) in week 6 of the development process.

**Project Diagrams**

**Use case Diagram:**

**Diagram

Description automatically generated**

**Software Process model for “AnyScript” Software:**

Depending on several factors such as project requirements, team size, team structure, customer involvement and project scope, we have chosen Agile Methodology. However, for our software “**AnyScript**” a platform that connects clients with talented writers who can produce high-quality copy or content within tight deadlines, the Scrum framework could be a good fit.

**Why we chose this software process model?**

Scrum is one of the most widely used Agile methodologies, and it is suitable for projects with a clear scope and specific deliverables. In Scrum, the project is divided into small sprints, typically two to four weeks, and the team delivers a potentially shippable product increment at the end of each sprint. This allows for frequent feedback from stakeholders and enables the team to adapt to changing requirements.

Overall, Scrum is a good fit for our software **AnyScript** because it provides a structured approach to development, allows for frequent feedback and adaptation, and places a lot of emphasis on customer involvement through the role of the Product Owner.

**Why our chosen process model different from other models?**

The plan-driven development is always planned, and the results are to be shown at the end of the product. While in agile development, planning is always incremental, where we can change the plan according to our customer requirements which is very important as we are engaging rapid clients or users.

Scrum, XP (Extreme Programming), and LSD (Lean Software Development) are all Agile methodologies, each with their strengths and weaknesses. Here are a few reasons why Scrum may be considered better than XP and LSD:

Scrum provides a more structured framework than XP or LSD, with clear roles, events, and artifacts that help to stay organized and focused on specific goals. This can be beneficial for larger or more complex projects like assessment software.

Scrum places a strong emphasis on customer collaboration and involvement throughout the development process, which ensures that the final product meets our customer's needs and expectations. While XP also emphasizes customer collaboration, LSD may not provide as much opportunity for customer involvement.

Most importantly, according to our chosen project Scrum is highly scalable and can be adapted to work with teams of any size, from small to large teams. While XP and LSD can also be scaled, they may require more effort to do so.

Scrum, XP, and LSD are all based on iterative and incremental development, which allows teams to deliver working software at regular intervals, providing feedback and ensuring that the project is on track. However, Scrum's more structured approach may be better suited to some teams and projects.

Scrum emphasizes continuous improvement, with regular retrospectives that help teams identify areas for improvement and implement changes. While XP and LSD also focus on continuous improvement, Scrum’s more structured framework may make it easier for us to implement changes.

Overall, Scrum is based on three pillars: Transparency, Inspection, and Adaptation hence we considered this best fitted model for our project.

1. **Project role identification and responsibilities:**

In a software project, there are typically several roles that are identified, each with their own set of responsibilities. These roles may vary depending on the size and complexity of the project, as well as the specific methodology being used. However, some common roles and their responsibilities in this software project are:

1. Project Manager: The project manager is responsible for planning, executing, and monitoring the project. They must ensure that the project is completed within budget, on schedule, and to the required quality standards. They also oversee communication between team members, stakeholders, and clients.
2. Software Architect: The software architect is responsible for designing the software system's architecture, including selecting the appropriate technologies, creating a high-level design, and ensuring that the design is scalable, maintainable, and secure.
3. Developer: Developers are responsible for writing and testing the software code. They work closely with the software architect and business analyst to ensure that the code meets the requirements and design specifications.
4. Quality Assurance (QA) Engineer: QA engineers are responsible for testing the software to ensure that it meets quality standards. They work with the development team to create test plans and test cases, execute tests, and report bugs.
5. DevOps Engineer: The DevOps engineer is responsible for deploying and managing the software application in production. They automate deployment processes, monitor the system, and troubleshoot issues that arise.
6. UX/UI Designer: The UX/UI designer is responsible for creating a user-friendly and visually appealing interface for the software system. They work with the business analyst to understand the user's needs and create wireframes and prototypes.

Overall, each role in a software project plays a critical role in ensuring that the project is successful. Clear communication, collaboration, and coordination between team members are essential for delivering a high-quality software product that meets the client's needs.Top of Form

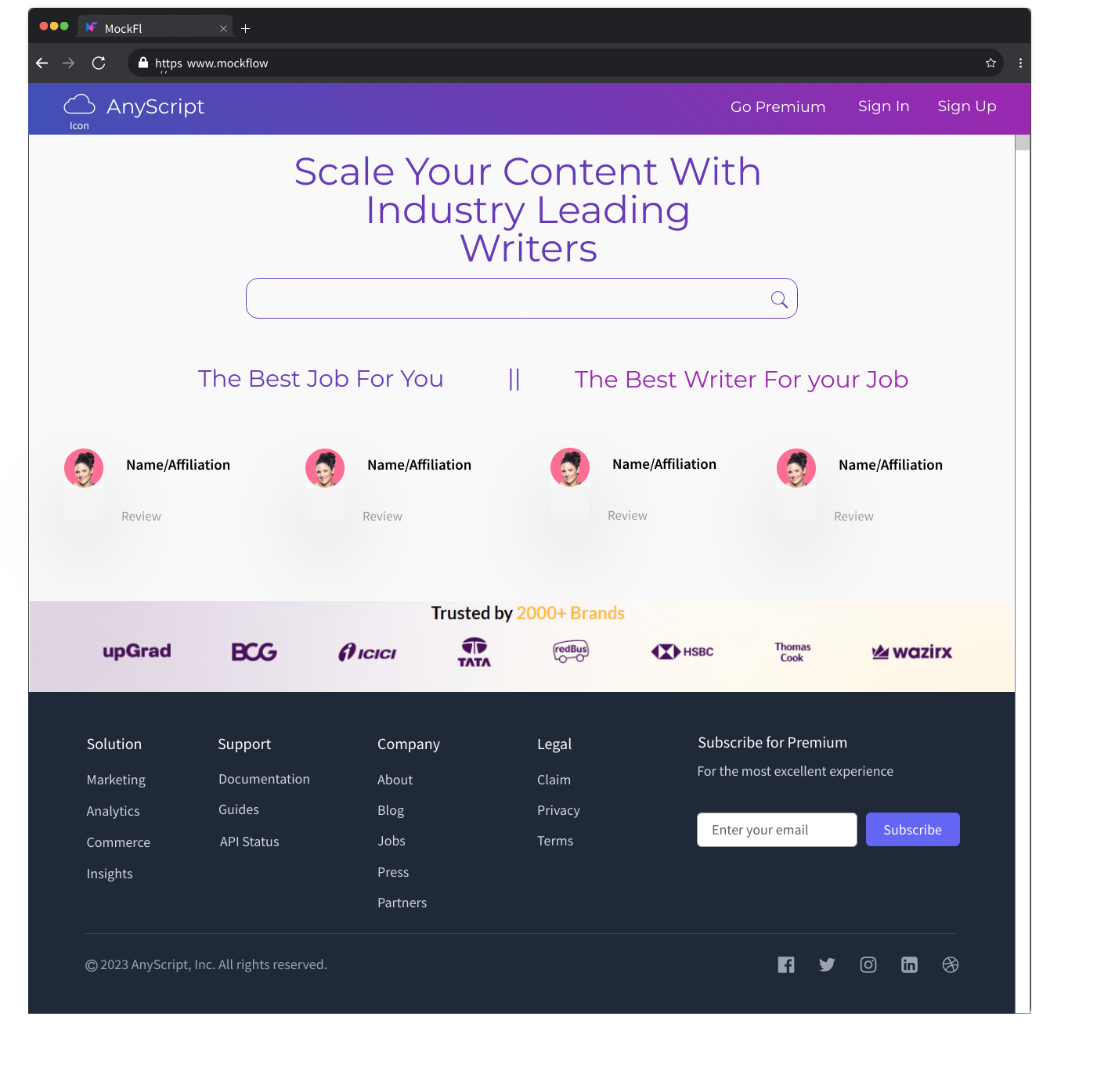
In addition to the roles and responsibilities of the project team, it's also important to consider the user roles in a software project. User roles represent the various types of users who will interact with the software product and have specific responsibilities and expectations. Some common user roles and their responsibilities in this software project are:

1. End-users: End-users are the people who will use the software product to perform specific tasks. Their responsibilities include learning how to use the software and providing feedback on its usability, functionality, and overall effectiveness.
2. Administrators: Administrators are responsible for managing the software product and its users. Their responsibilities include setting up user accounts, assigning user roles and permissions, and configuring the software to meet the needs of their organization. They will also work as support staff.
3. Customers: Customers are the people who will purchase and use the software product. Their responsibilities include providing feedback on the software's features, pricing, and overall value proposition. They may also be responsible for making purchasing decisions and negotiating contracts with the software vendor.
4. Testers: Testers are responsible for testing the software product to ensure that it meets the user's requirements and works as expected. Their responsibilities include creating and executing test cases, reporting bugs, and verifying that bugs have been fixed.

By understanding the various user roles in a software project, the development team can design and develop a software product that meets the needs of its intended users. The user roles can also help guide decisions related to software features, user interface design, and overall usability.

**Wireframe Design**

1. **Loading Page**



1. Sign In

Graphical user interface, application, Teams

Description automatically generated

1. **Premium Subscription**

**Graphical user interface, application, website

Description automatically generated**

1. Personal Profile

Graphical user interface, application

Description automatically generated

1. Search Jobs/Writers

Graphical user interface, application, Teams

Description automatically generated

1. Other’s Profile view

A screenshot of a computer

Description automatically generated

Project Test Planning for “AnyScript”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Name: Anyscript | | | Test Designed By: ASQA Team | |
| Test Case ID: FR\_1 | | | Test Designed Date: April 2, 2023 | |
| Test Priority: High | | | Test Executed By: ASQA Team | |
| **Module Name: Registration** | | | Test Execution Date: April 2, 2023 | |
| **Test Title: Verify Register of new account** | | | | |
| Description: Verify that the website login page allows registered users to log in with their valid username and password, and displays the user account home page. | | | | |
| Precondition: User must have a valid username and password. | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status |
| 1. Navigate to the registration page. 2. Fill in the required fields such as name, email, password, and confirm password. 3. Click on the submit button. 4. Verify that a confirmation message is displayed indicating that the registration was successful. 5. Check the registered email account for a verification link or code. 6. Click on the verification link or enter the code provided in the email to complete the registration process. 7. Verify that the user is redirected to the login page and can successfully log in using the registered email and password. | Username: testuser123  Email:  Testuser@gmail.com  Password: Test123! | Confirmation message comes and the verification code is sent to email.  Upon confirmation the id gets registered and login ready | As Expected | Pass |
| Post Condition: User is stored in the database encrypted | | | | |

Table 1: Registration Test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Name: Anyscript | | | Test Designed By: ASQA Team | |
| Test Case ID: FR\_2 | | | Test Designed Date: April 2, 2023 | |
| Test Priority: High | | | Test Executed By: ASQA Team | |
| **Module Name: Login** | | | Test Execution Date: April 2, 2023 | |
| **Test Title: Verify Login with Valid Username and Password** | | | | |
| Description: Verify that the website login page allows registered users to log in with their valid username and password, and displays the user account home page. | | | | |
| Precondition: User must have a valid username and password. | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status |
| 1. Go to the website 2. Enter username 3. Enter password 4. Click submit | Username: testuser123  Password: Test123! | User should login into the application | As Expected | Pass |
| Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database. | | | | |

Table 2: Valid Login Test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Name: Anyscript | | | Test Designed By: ASQA Team | |
| Test Case ID: FR\_3 | | | Test Designed Date: April 2, 2023 | |
| Test Priority: High | | | Test Executed By: ASQA Team | |
| **Module Name: Login** | | | Test Execution Date: April 2, 2023 | |
| **Test Title: Verify Login with invalid Username and Password** | | | | |
| Description: Verify that the website login page disallows unregistered users to log in with their invalid username and password, and displays invalid account. | | | | |
| Precondition: User must not have a valid username and password. | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status |
| 1. Go to the website 2. Enter username 3. Enter password 4. Click submit | Username: testuser12  Password: Test12! | User should not login into the application | As Expected | Pass |
| Post Condition: User is invalidated with database. | | | | |

Table 3: Invalid Login Test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Name: Anyscript | | | Test Designed By: ASQA Team | |
| Test Case ID: FR\_4 | | | Test Designed Date: April 2, 2023 | |
| Test Priority: High | | | Test Executed By: ASQA Team | |
| **Module Name: Job Posting** | | | Test Execution Date: April 2, 2023 | |
| **Test Title: Verify Job Posting** | | | | |
| Description: Verify that the website is posting job as expected | | | | |
| Precondition: User must have a valid user id | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status |
| 1. Navigate to the job posting page 2. Fill out the job posting form with valid data 3. Click "Submit" 4. Verify that the job is posted on the platform 5. Log out of the client account and log in as a writer 6. Navigate to the available jobs page 7. Verify that the posted job is displayed | Job Title:  Content Writer  Job Description:  Write a 1000 word article on travel tips  Budget:  $50  Deadline:  2023-04-10 | posted job is displayed | As Expected | Pass |
| Post Condition: Store Job posted in database | | | | |

Table 4: Job Posting Test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Name: Anyscript | | | Test Designed By: ASQA Team | |
| Test Case ID: FR\_5 | | | Test Designed Date: April 2, 2023 | |
| Test Priority: High | | | Test Executed By: ASQA Team | |
| **Module Name: Job Apply** | | | Test Execution Date: April 2, 2023 | |
| **Test Title: Verify applying for a job** | | | | |
| Description: Verify that a posted job can be applied for. | | | | |
| Precondition: User must have a valid user id. | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status |
| 1. Navigate to the job posting page. 2. Click on the "Apply Now" button. 3. Fill out the job application form. 4. Attach required documents (e.g., resume, cover letter). 5. Submit the application. 6. Verify that the application was submitted successfully. | User credentials (username and password) for logging in  Resume file: name\_resume.pdf  Personal Message/Letter | The client received the notification and the message | As Expected | Pass |
| Post Condition: Store Job apply in database | | | | |

Table 5: Job Acceptance Test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Name: Anyscript | | | Test Designed By: ASQA Team | |
| Test Case ID: FR\_6 | | | Test Designed Date: April 2, 2023 | |
| Test Priority: Medium | | | Test Executed By: ASQA Team | |
| **Module Name: Messaging** | | | Test Execution Date: April 2, 2023 | |
| **Test Title: Verify message receive and delivery** | | | | |
| Description: Verify message can be received and be delivered | | | | |
| Precondition: User must have a valid user id | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status |
| 1. Open the messaging 2. Select a contact or create a new conversation. 3. Type a message in the message box. 4. Click the send button. 5. Wait for the message to be delivered. 6. Check if the message has been delivered to the recipient. 7. Wait for the recipient to read the message. 8. Check if the recipient has read the message. 9. Do the exact same from the other side to check | Message:  Test Message\_1  Test send file: name\_resume.pdf  Test voice  Test call | The sender and the recipient delivered and received the accurate message | As Expected | Pass |
| Post Condition: Store all message in database encrypted | | | | |

Table 6: Messaging Test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Name: Anyscript | | | Test Designed By: ASQA Team | |
| Test Case ID: FR\_7 | | | Test Designed Date: April 2, 2023 | |
| Test Priority: High | | | Test Executed By: ASQA Team | |
| **Module Name: Payment** | | | Test Execution Date: April 2, 2023 | |
| **Test Title: Verify Payment process** | | | | |
| Description: Verify payment is sent to the writer from the client | | | | |
| Precondition: User must have a valid user id and apply and deliver a job | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status |
| 1. Launch the payment page by clicking on the payment option 2. Verify that the payment page is secure and the payment options are displayed correctly 3. Select the payment method and enter the required details (e.g. credit card number, CVV, expiry date) 4. Verify that the payment details are correct and click on the "Pay" button 5. Verify that the payment is processed successfully and the user is redirected to a confirmation page 6. Verify that the payment confirmation page displays the details of the payment (e.g. payment amount, payment method, transaction ID) 7. Verify that the payment confirmation email is sent to the user's email address with the payment details | Valid credit card details (number, CVV, expiry date)  Invalid credit card details (number, CVV, expiry date)  Various payment methods (e.g. Visa, Mastercard, PayPal)  Payment amounts (e.g. $10, $50, $100) | Payment received through valid payment credentials | As Expected | Pass |
| Post Condition: Job has been removed from the posted list in the database, job done by the writer is added | | | | |

Table 7: Payment Test

**Work Breakdown Structure**

**A picture containing text, diagram, plan, parallel

Description automatically generated**

**Constructive cost Model (*COCOMO*):**

Let’s assume Source Line of Code is 4000.

So, effort need to be, PM = 2.4 (4000/1000)1.05 = 10.289

Development time, DM = 2.5 \* (PM)0.38 = 6.0623 = 6

Required number of people, ST = PM/DM = 1.697 = 2

**That means we need to work for (4\*6) =24 weeks.**

**9. Timeline Chart (Project Plan)-1**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Pregame Phase | | | | | | Development | | | | | | | | | | | | Postgame Phase | | | | | |
| Planning | | | Architecture | | | Sprint1 | | | | Sprint2 | | | | Sprint3 | | | |
| Weeks  Person | Week1 | Week2 | Week3 | Week4 | Week5 | Week6 | Week7 | Week8 | Week9 | Week10 | Week11 | Week12 | Week13 | Week14 | Week15 | Week16 | Week17 | Week18 | Week19 | Week20 | Week21 | Week22 | Week23 | Week24 |
| A:T |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| B:T |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| C:M&T |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| D: M&T |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| E: M&T |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| F: M&T |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| G:M |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| H:T |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| I:M |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Here, M Mahedi & T means Tazrif (As staffing necessary is 2)

A: Project initiation, Scoping, requirements gathering and planning

B: Design, product backlog creation and sprint planning

C: Requirements for each sprint

D: Analysis for each sprint

E: Development for each sprint

F: Testing for each sprint

G: Integration testing

H: System testing

I: Release preparation and launch

**10. EVA Analysis:**

|  |  |  |
| --- | --- | --- |
| Task | Planned effort | Actual effort |
| 1 | 10 | 11 |
| 2 | 8 | 10 |
| 3 | 7 | 5 |
| 4 | 9 | 7 |
| 5 | 7.5 | 6 |
| 6 | 4 | 7 |
| 7 | 14 | 11 |
| 8 | 6 | 7 |
| 9 | 9.5 | 10.5 |
| 10 | 8.5 | 10 |
| 11 | 6 | --- |
| 12 | 10 | --- |
| 13 | 5 | --- |
| 14 | 8 | --- |
| 15 | 6 | --- |

Given Total Task=54

Effort Estimated=309

BAC=309

SPI=BCWP/BCWS=83.5/118.5=0.704641

SV=BCWP-BCWS=83.5-118.5=-35 person-day

CPI=BCWP/ACWP=83.5/84.5=0.99

CV=BCWP-ACWP=83.5-84.5=-1 person-day

% schedule for completion=BCWS/BAC=(118.5/309)\*100%=38.34%

% complete=BCWP/BAC=(83.5/309)\*100%=27.02%.

**11. Timeline Chart-2**

**Pre-Game Phase:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Work Task | Week 1 | | | | | Week 2 | | | | | Week 3 | | | | | Week 4 | | | | | Week 5 | | | | | Week 6 | | | | |
| Project scope is defined |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Requirements are gathered |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Project plan is created |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Project team is built |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| User stories are developed |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Product backlog is created |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Sprint planning is conducted |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Game Phase:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Work Task | Sprint 1 | | | | | | | | | | Sprint 2 | | | | | | | | | | Sprint 3 | | | | | | | | | |
| Week 7 & 8 | | | | | Week 9 & 10 | | | | | Week 11 & 12 | | | | | Week 13 & 14 | | | | | Week 15 & 16 | | | | | Week 17 & 18 | | | | |
| Login features is developed |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Data input feature is developed |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Data validation feature is developed |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Report generation feature is developed |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Data export feature is developed |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Payment feature is developed |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Post-Game Phase:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Work Task | Week 19 | | | | | Week 20 | | | | | Week 21 | | | | | Week 22 | | | | | Week 23 | | | | | Week24 | | | | |
| Sprint review is conducted |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Sprint retrospective is conducted |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| User acceptancy testing is conducted |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Software is deployed |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| User training is provided |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Post release review is conducted |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**12. Risk Management Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risks** | **Category** | **Probability** | **Impact** | **RMMM** |
| Size estimate maybe significantly low | PS | 70% | 2 |  |
| Larger number of users than planned | PS | 20% | 3 |  |
| Less reuse than planned | PS | 50% | 2 |  |
| End user resists system | BU | 30% | 2 |  |
| Delivery deadline will be tightened | BU | 75% | 1 |  |
| Funding will be lost | CU | 30% | 1 |  |
| Customer will change requirements | PS | 25% | 2 |  |
| Technology will not meet expectations | TE | 20% | 1 |  |
| Lack of Training on tools | DE | 50% | 3 |  |
| Staff inexperienced | ST | 30% | 3 |  |
| Staff turnover will be high | ST | 60% | 2 |  |
| Personnel shortfalls | ST | 50% | 2 |  |
| Unrealistic time and cost estimates | BU | 35% | 1 |  |
| Developing the wrong software functions | PR | 17% | 1 |  |
| Developing the wrong interface | PR | 20% | 2 |  |
| Gold plating | PR | 30% | 2 |  |
| Late changes to requirements | CU | 60% | 2 |  |
| Shortfalls in externally performed tasks | DE | 40% | 3 |  |
| Real time performance problems | ST | 45% | 3 |  |
| Development technically too difficult | TE | 60% | 2 |  |
| Data loss | DE | 45% | 1 |  |
| Handling sensitive information | DE | 40% | 2 |  |
| Delayed software development due to lack of resources | DE | 20% | 3 |  |
| Intellectual property infringement | BU | 30% | 4 |  |
| Data integrity issues due to modifications in software | PS | 30% | 2 |  |
| Slow response due to large software | PS | 60% | 4 |  |
| Maintenance and upgrade difficulties of large software | PS | 50% | 3 |  |
| Cost overrun due to ineffective communication with customer | CU | 10% | 3 |  |
| Insufficient user training resulting in user errors | CU | 40% | 3 |  |
| User client payment defaults | BU | 10% | 1 |  |
| Data breach for the software | TE | 30% | 1 |  |
| Data loss due to hardware or software failure | TE | 10% | 1 |  |
| Compliance with labor and tax regulations of different jurisdictions | BU | 10% | 2 |  |
| Increased technical debt due to constraints imposed by management | BU | 10% | 4 |  |
| Lack of stake holders | BU | 30% | 2 |  |
| Integration issues with third party systems | BU | 70% | 3 |  |
| Inadequate communication leading to misunderstanding | ST | 40% | 3 |  |
| Sudden change of team leading to confusions and conflicts | ST | 20% | 2 |  |
| Poor software quality that does not meet the user’s needs | PR | 30% | 2 |  |
| Budget overrun due to unforeseen expenses | PR | 70% | 3 |  |
| Lack of consistency of the development organization to meet regulatory or compliance requirements | PR | 10% | 3 |  |

**Impact values:**

1. Catastrophic
2. Critical
3. Marginal
4. Negligible