**PRACTICAL - 4**

**AIM:** Study and apply stepwise project planning activities

**Scenario:** Airbnb and Farfetch.

**Task:**

1. Study the steps and activities involved in project planning.
2. Create project planning for a given scenario

**Team Details :**

| **Sr. No.** | **Name** | **Enrollment No.** |
| --- | --- | --- |
| **Team Leader** | **Fenil Shilodre** | **202203103510041** |
| **Team Member 1** | **Gati Shah** | **202203103510261** |
| **Team Member 2** | **Angat Shah** | **202203103510097** |
| **Team Member 3** | **Yash Patel** | **202203103510228** |
| **Team Member 4** | **Sarth Chaudhari** | **202303103510106** |

**Project Title : Airbnb System**

1. Identify Project Scope and Objectives

1. Identify Objectives and Measures of Effectiveness in Meeting Them:
   * Objective:
     + Provide a seamless booking experience for users worldwide.
     + Enable property hosts to efficiently list and manage properties.
     + Ensure secure and fast transactions with integrated payment systems.
     + Support multiple languages and currencies for a global user base.
   * Effectiveness Measure:
     + User satisfaction through surveys and feedback.
     + Increased number of bookings.
     + Platform uptime and performance metrics.
     + Successful processing of payments and security compliance.
2. Establish Project Authority:
   * Project manager (Team Leader) is responsible for the oversight, while technical leads will oversee specific modules (frontend, backend, database, security)
3. Identify Stakeholders:
   * Airbnb users (guests)
   * Hosts providing experiences
   * Airbnb operations and customer support teams
   * Marketing and Sales teams
   * Technology/Development team
4. Modify Objectives in Light of Stakeholder Analysis:
   * Adjustments for local cultural nuances in booking experiences based on different regions.
5. Establish Methods of Communication with All Parties:
   * Weekly updates with the team via Slack and bi-weekly meetings for detailed progress reviews.

#### 2. Identify Project Infrastructure

1. Establish Relationship Between Project and Strategic Planning:
   * The integration of experiences directly supports Airbnb’s strategy to diversify offerings and increase customer engagement.
2. Identify Installation Standards and Procedures:
   * The feature will be built using Airbnb’s existing tech stack (React, Node.js, AWS).
   * Standard procedures will involve using a CI/CD pipeline for updates and testing.
3. Identify Project Team Organization:
   * The project team is organized into sub-teams: development, UX/UI, marketing, and testing.

#### 3. Analyze Project Characteristics

1. Distinguish the Project as Either Objective-Driven or Product-Driven:
   * The project is **product-driven**, as it focuses on introducing a new feature for Airbnb users (experience bookings).

1. Analyze Other Project Characteristics:
   * The project will be phased, starting with a beta test in a limited region before a global rollout.
2. Identify High-Level Project Risks:
   * Risk of low adoption if users do not see value in the experience booking feature.
   * Potential integration issues between the accommodation and activity booking systems.
3. Take into Account User Requirements Concerning Implementation:
   * The platform needs to be easy to use, with the ability to filter experiences by location, type, and rating.
4. Select General Life-Cycle Approach:
   * The project will use an **Agile approach**, with sprints lasting 2 weeks, ensuring continuous feedback and improvement.
5. Review Overall Resource Estimates:
   * Estimated resources: 4 developers, 2 UX/UI designers, 1 QA tester, 2 marketing professionals. Also include server infrastructure, cloud services and third-party integrations

#### 4. Identify Project Products and Activities

1. Identify and Describe Project Products (Including Quality Criteria):
   * Product 1: Web and mobile application
     + Quality criteria: User interface for booking, profile management.
   * Product 2: Payment integration system.
     + Quality criteria: Seamless integration, intuitive UI/UX, reliable transaction process.
2. Document Generic Product Flows:
   * Flow 1: User logs in → searches for properties → filters results → makes a booking → payment → receives confirmation.
   * Flow 2: Host lists a property → manages listing → receives booking requests → communicates with guests.

1. Recognize Product Instances:
   * Different product instances: Activity categories such as different user profiles (traveler, host, admin) or booking instances (new, canceled, confirmed).
2. Produce Ideal Activity Network:
   * Ideal activity flow: Design → Development → Testing → Deployment → Maintenance
   * Activity flow includes system architecture, UI/UX design, database creation, and integration
3. Modify Ideal to Take into Account Need for Stages and Checkpoints:
   * Critical checkpoints: Planning → Design → Development → Testing → Deployment → Feedback → Maintenance.

#### 5. Estimate Effort for Each Activity

1. Carry Out Bottom-Up Estimates:
   * Estimation based on prior similar feature builds:
     + Front-end development: 120 hours
     + Back-end development: 150 hours
     + UI/UX design: 100 hours
     + Marketing and promotion: 80 hours
2. Revise Plan to Create Controllable Activities:
   * The activities are broken into smaller tasks, each reviewed during sprint retrospectives.

#### 6. Identify Activity Risks

1. Identify and Quantify Activity-Based Risks:
   * Risk 1: Integration issues with current Airbnb backend (impact: medium).
   * Risk 2: Low user engagement (impact: high).
2. Plan Risk Reduction and Contingency Measures Where Appropriate:
   * Develop a contingency plan to enhance the user interface if engagement is low (e.g., additional features like instant booking confirmation).
3. Adjust Plans and Estimates to Take Account of Risks:
   * Allocate additional time for troubleshooting potential backend issues.

#### 7. Allocate Resources

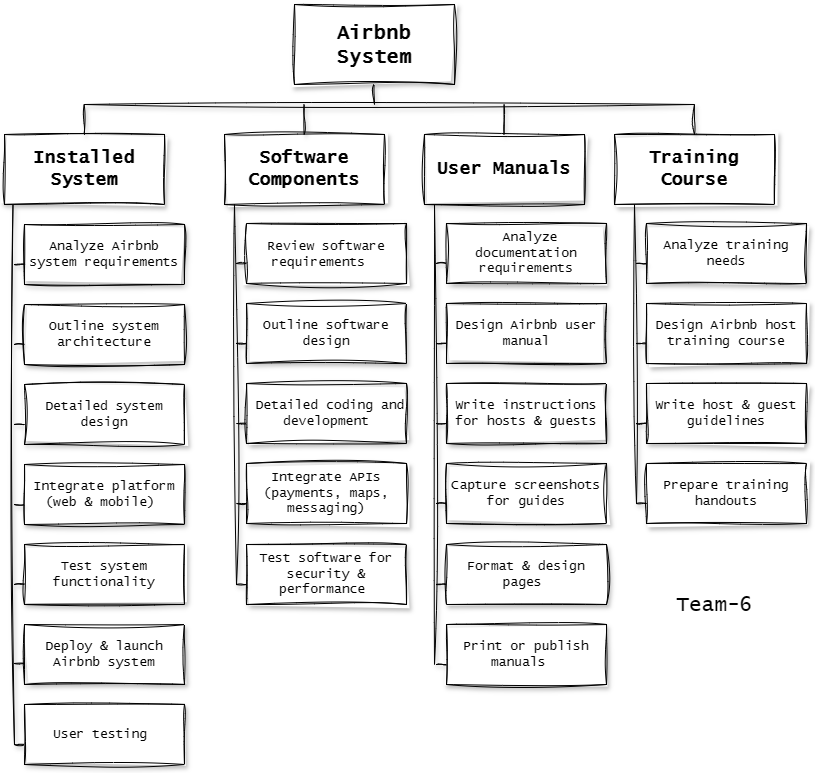
1. Identify and Allocate Resources:
   * Development: 4 developers, 1 QA tester
   * Design: 2 UX/UI designers
   * Marketing: 2 professionals for outreach and content creation
2. Revise Plans and Estimates to Take Account of Resource Constraints:
   * Ensure that the resources allocated to each team are sufficient to meet deadlines while considering potential delays in the design phase.

#### 8. Review/Publicize Plan

1. Review Quality Aspects of Project Plan:
   * Ensure that the feature is user-friendly, scalable and adheres to Airbnb’s standards of accessibility and security.
2. Document Plans and Obtain Agreement:
   * Present the plan to Airbnb stakeholders (e.g., product managers and executives) for approval.

#### 9/10. Execute Plan/ Lower Levels of Planning

* + Begin development and design sprints, prioritizing core features.
  + Break down the tasks into smaller sub-tasks, assigning them to specific team members.



Work Breakdown Structure for AirBnb System

**Project Title : Farfetch E-commerce System**

#### 1. Identify Project Scope and Objectives

1. Identify objectives and measures of effectiveness in meeting them:

* + Objectives:
    - Create an intuitive and user-friendly platform for luxury fashion shopping.
    - Enable boutiques and brands to list and manage their products.
    - Ensure secure, reliable payment processing and multiple payment options.
    - Provide personalized product recommendations based on customer browsing behavior and preferences.
  + Measures of effectiveness:  
    - Customer satisfaction: Measure through customer feedback, reviews, and ratings.
    - Conversion rate: Track the percentage of visitors who complete purchases.
    - Platform uptime: Ensure availability and responsiveness of the platform at all times.
    - Order fulfillment: Monitor delivery times and success rates for global shipping.

2. Establish a project authority:

* + The Project Manager (PM) will oversee the entire project.
  + Technical Leads: Oversee various modules (Frontend, Backend, Payment Systems, etc.).
  + Stakeholders: Users (buyers, boutiques, brands), payment gateways, delivery services, admins.

3. Identify stakeholders:

* + Primary stakeholders:
    - Customers (shoppers who buy luxury fashion products).
    - Boutiques and Brands (sellers listing products).
    - Admins (platform management and oversight).
  + Secondary stakeholders:
    - Payment gateway providers (Stripe, PayPal, etc.).
    - Shipping and logistics partners (FedEx, DHL).
    - Third-party APIs for inventory, recommendations, etc.
    - Compliance and regulatory authorities (GDPR, PCI-DSS).

4. Modify objectives in light of stakeholder analysis:

* + Consider region-specific shipping regulations and payment methods.
  + Enhance personalized recommendations based on data analytics to better match products with customer preferences.

5. Establish methods of communication with all parties:

* + Internal communication: Slack or Microsoft Teams for team collaboration.
  + Stakeholder meetings: Weekly Zoom meetings to discuss project progress, updates, and risks.
  + Project management tools: Jira or Trello to track tasks, milestones, and progress.

#### 2. Identify Project Infrastructure

1. Establish relationship between project and strategic planning:

* + The project supports Farfetch’s business goal of increasing global sales in the luxury fashion market and providing a seamless customer experience.
  + Global scalability: The system must handle a large number of international users, transactions, and product inventories.

2. Identify installation standards and procedures:

* + Cloud infrastructure (AWS, Google Cloud) to handle scaling and ensure high availability.
  + Multi-platform development: Web and mobile app (React Native for iOS and Android).
  + Continuous integration/continuous delivery (CI/CD) for regular updates and testing.

3. Identify project team organization:

* + Project Manager (PM): Leads the project, handles stakeholder communication, ensures deadlines.
  + Technical Leads:
    - Frontend Lead (UI/UX design for web and mobile).
    - Backend Lead (server-side functionality, database, API design).
    - Security Lead (payment security, data privacy).
  + Developers: Full-stack and mobile developers.
  + QA Engineers: Ensuring the platform is bug-free and secure.
  + DevOps: Managing cloud infrastructure, scaling, and deployment.

#### 3. Analyze Project Characteristics

1. Distinguish the project as either objective- or product-driven:

* + Product-driven: The goal is to develop the Farfetch platform where users can shop luxury items and boutiques can list products.

2. Analyze other project characteristics:

* + Medium to large-scale with a global audience, handling high traffic and transactions.
  + Third-party integrations: Payment gateway, shipping services, and recommendation engines.
  + Real-time data: Personalized recommendations, live inventory updates.
  + High availability and scalability required due to international users.

3. Identify high-level project risks:

* + Data breaches: Potential security risks involving user data and payment details.
  + Payment failures or fraud during transactions.
  + Shipping delays or issues with international deliveries.
  + Integration issues: Problems integrating payment gateways or shipping APIs.

4. Take into account user requirements concerning implementation:

* + Functional Requirements:
    - Easy navigation for product search and purchase.
    - Secure payment gateway integration.
    - Global shipping and order tracking.
    - Personalized product recommendations.
  + Non-functional Requirements:
    - Scalability to handle traffic surges, especially during sales events.
    - High availability to ensure the system is up and running 24/7.
    - Data security to protect customer information and prevent fraud.

5. Select general life-cycle approach:

* + **Hybrid Agile** Approach: Iterative development, allowing flexibility to adjust features based on feedback.
  + Scrum framework for sprint-based development and regular updates.

6. Review overall resource estimates:

* + Personnel: Developers, designers, security experts, DevOps, QA engineers.
  + Technological resources: Cloud infrastructure, payment gateway APIs, shipping service APIs.
  + Financial resources: Budget for cloud services, third-party integrations, marketing campaigns.

#### 4. Identify Project Products and Activities

1. Identify and describe project products (including quality criteria):

* + Products:
    - E-commerce website for browsing and purchasing products.
    - Payment system integration (secure payments via Stripe, PayPal).
    - Product catalog with detailed descriptions, images, and prices.
    - Admin Panel for managing products, users, and orders.
    - Order management system with real-time order tracking.
  + Quality Criteria:
    - User experience: Easy navigation, fast load times, intuitive design.
    - Security: Secure payment processing, encrypted user data.
    - Scalability: Ability to handle large numbers o f users and transactions.

2. Document generic product flows:

* + Customer Flow: Login → Browse Products → Add to Cart → Checkout → Payment → Order Confirmation.
  + Admin Flow: Login → Manage Products → Review Orders → Manage Users.

3. Recognize product instances:

* + User Types: Customers, Admins.
  + Order Status: Pending, Processed, Shipped, Delivered.

4. Produce ideal activity network:

* + Design → Development → Testing → Deployment → Maintenance.

5. Modify ideal to take into account need for stages and checkpoints:

* + Phase 1: Requirements gathering and design (UI/UX, database).
  + Phase 2: Backend and frontend development.
  + Phase 3: API integration (payment, shipping).
  + Phase 4: Testing and quality assurance.
  + Phase 5: Deployment and monitoring.
  + Phase 6: Ongoing maintenance and updates.

#### 5. Estimate Effort for Each Activity

1. Carry out bottom-up estimates:

* + Estimate the time required for each module, such as payment gateway integration, product catalog setup, and user authentication.

2. Revise plan to create controllable activities:

* + Break tasks into manageable sprints (e.g., sprint 1 for product catalog, sprint 2 for payment integration).

#### 6. Identify Activity Risks

1. Identify and quantify activity-based risks:

* + Integration risk: Failure of payment gateway or shipping API.
  + Security risk: Data breaches or fraud during payment transactions.

2. Plan risk reduction and contingency measures:

* + Implement multi-factor authentication (MFA) for payment processing.
  + Use encryption for sensitive data (credit card numbers, addresses).

3. Adjust plans and estimates to take account of risks:

* + Add buffer time for third-party API integration and security testing.

#### 7. Allocate Resources

1. Identify and allocate resources:

* + Personnel: Developers, security specialists, designers, QA engineers.
  + Technology: Cloud infrastructure, payment gateways, shipping APIs.

2. Revise plans and estimates to take account of resource constraints:

* + Adjust plans to ensure timely integration of critical systems, such as the payment gateway.

#### 8. Review/Publicize Plan

1. Review quality aspects of project plan:

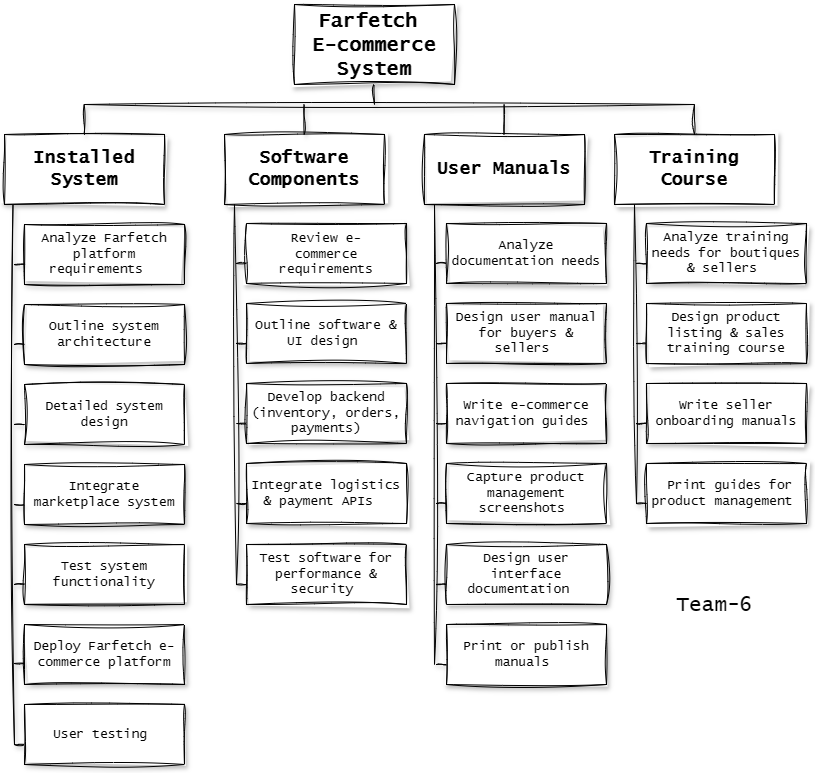
* + Ensure all quality aspects (security, scalability, performance) are adequately addressed.

2. Document plans and obtain agreement:

* + Stakeholder sign-off for the project plan to ensure alignment with business goals.

#### 9/10. Execute Plan/Lower Levels of Planning

* Execution:
  + Sprint 1: Set up cloud infrastructure and begin UI/UX design.
  + Sprint 2: Develop product catalog, integrate basic features.
  + Sprint 3: Complete payment system integration, begin testing.



Work Breakdown Structure for FarFetch