Capstone Project

Lecture 2

Introduction to Project Planning and Management

- Definition: What is Project Planning?
  - A process to define objectives, scope, and steps to achieve project goals.
- Importance:
  - Ensures projects are completed on time, within scope, and on budget.
- Key Components:
  - Project scope
  - Timeline
  - Resources
  - Risk management

Key Components of Project Planning

- **Project Scope:** Defining the boundaries of the project, deliverables, and objectives.

- **Resource Management:** Identifying and allocating resources (human, financial, and material).

- **Timeline Creation:** Developing a schedule with milestones and deadlines.

- **Risk Management:** Identifying potential risks and creating mitigation strategies.

- **Stakeholder Management:** Engaging and communicating with stakeholders.

Agile Project Planning

- Definition: An iterative approach to project management and software development that focuses on continuous improvement and flexibility.

- Core Principles:

- Customer collaboration

- Responding to change

- Frequent delivery of working software

- Individuals and interactions over processes and tools

Scrum Framework in Agile

- Overview: A subset of Agile, focusing on iterative and incremental progress through Sprints.
- Key Roles:
  - Product Owner
  - Scrum Master
  - Development Team
- Key Events:
  - Sprint Planning
  - Daily Stand-ups
  - Sprint Review
 - Sprint Retrospective

Scrum Process Flow

- Visual Representation:

- Diagram showing the flow from Sprint Planning to Retrospective

- Explanation: The Scrum process helps in adapting to changes quickly and delivering value faster.

Development in Agile

- Continuous Integration: Regularly integrating code changes to detect issues early.

- Test-Driven Development (TDD): Writing tests before code to ensure functionality.

- Automation: Using tools to automate repetitive tasks like testing and deployment.

SCRUM PROCESS
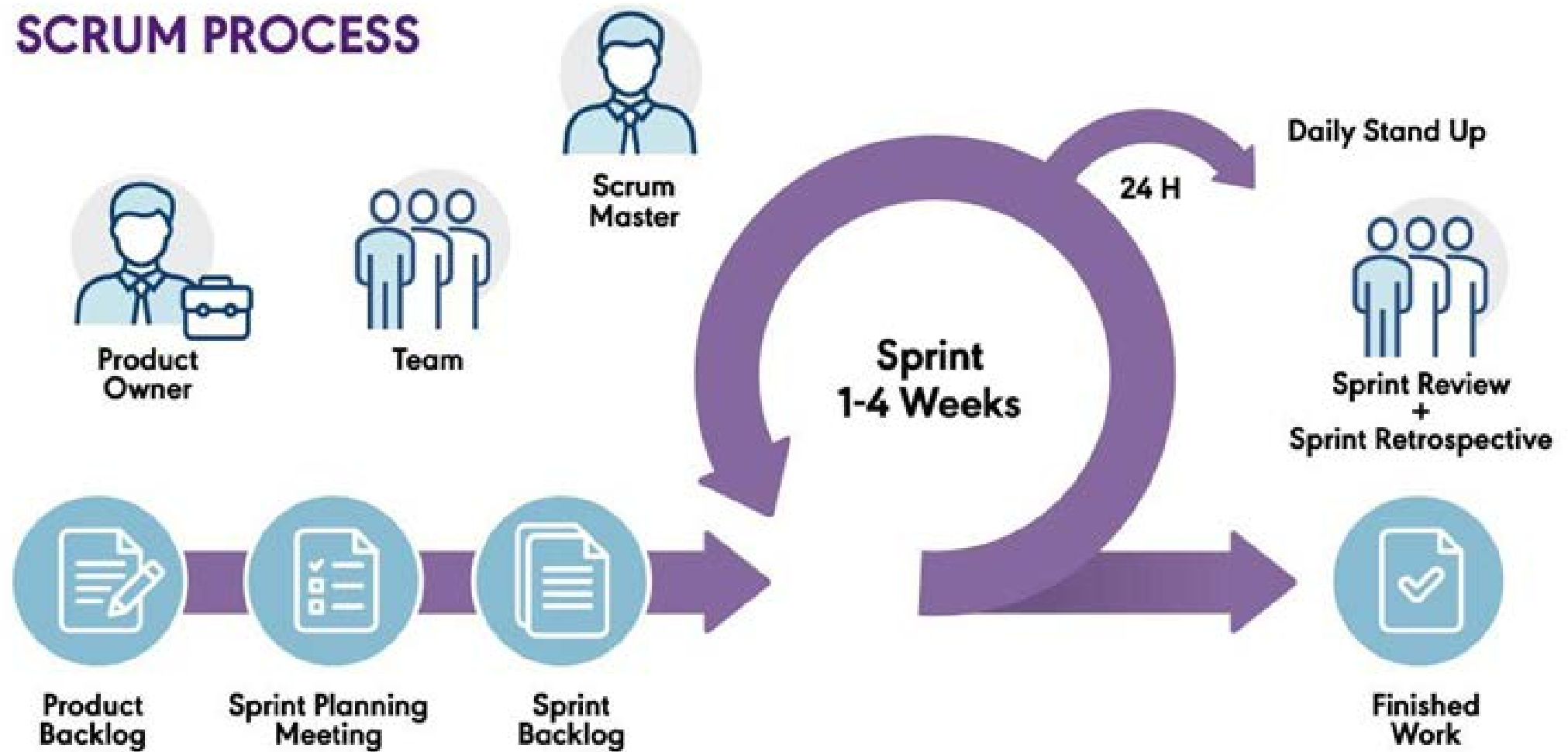
Capstone Project

- Project: Appointment Booking App for a Health Clinic
- Objective: Develop a functional web application for managing appointments and schedules.
- Key Features:
  - User management (e.g., patient and clinic staff accounts)
  - Appointment scheduling (e.g., booking, rescheduling, cancellation)
  - Notifications and reminders (e.g., appointment confirmations)
  - Reporting and analytics (e.g., appointment statistics)

Project Planning and Management

- Scope Definition: Defining the project's scope to include features like user authentication, appointment booking, and doctor availability management.

- Resource Allocation: e.g. Assign front-end tasks (e.g., UI design using React) to team members with experience in web development.

- Tools:
  - Slack/Microsoft Teams: For team communication and coordination.
  - Trello/Jira: For task tracking and backlog management.
  - Google Docs: For collaborative project documentation.

Task Breakdown and Timeline

- Task Breakdown: Breaking down the project into tasks such as setting up the database schema, developing the front-end, and integrating the API.
- Tools:
  - Asana/Monday.com: For creating and assigning tasks.
  - GanttProject/TeamGantt: For creating Gantt charts.
- Gantt Chart Example / Demo:
  - Example: A Gantt chart showing the timeline for developing user authentication, appointment scheduling, and notifications features.

Risk Management

- Potential Risks:
  - Example: Risk of scope creep if additional features (e.g., integration with electronic health records) are requested mid-project.
- Tools:
  - Risk Register in Excel/Google Sheets: To document and track risks.
  - Confluence: For centralized project documentation, including risk management plans.
- Mitigation Strategies:
  - Example: Regular team meetings to reassess priorities and scope.

Implementing Scrum in the Project

- Sprint Planning:
  - Example: Plan a sprint focused on developing and testing the user interface for appointment scheduling.
- Tools:
  - Jira/Scrum Boards: For sprint planning and tracking.
- Daily Stand-ups:
  - Example: Use daily stand-ups to discuss progress on integrating the front-end with the back-end API for booking appointments.
- Sprint Review and Retrospective:
  - Example: Review the sprint outcomes, such as the implementation of appointment reminders, and discuss improvements.
- Tools:
  - Zoom/Google Meet: For virtual stand-up meetings and sprint reviews.

Development Phases in the Project

- Front-End Development: E.g. Designing a user-friendly interface that allows patients to easily book and manage appointments.
- Tools:
  - React.js/Bootstrap: For front-end development.
  - Figma/Adobe XD: For UI/UX design.
- Back-End Development: - E.g. Setting up a secure server to handle patient data and appointment bookings.
- Tools:
  - Node.js/Django: For back-end development.
  - Postman: For testing API endpoints.
- Integration: Example - Ensure that the front-end can seamlessly communicate with the back-end to display available appointment slots.
- Tools: Swagger- For documenting and testing RESTful APIs.

Testing and Quality Assurance

- Unit Testing: E.g. Test individual components, such as the appointment booking function, to ensure they work correctly.
- Tools:
  - Jest/Mocha: For front-end and back-end unit testing.
  - Selenium/Cypress: For end-to-end testing.
- Integration Testing: E.g. Test the interaction between the appointment scheduling front-end and the back-end database.
- User Acceptance Testing (UAT): E.g. Gather feedback from users on the ease of booking and rescheduling appointments.
- Tools:
  - Google Forms/SurveyMonkey: For collecting user feedback.
  - Bugzilla: For tracking bugs and issues found during testing.