**Movie Booking Website**

Mohammad Jouni - 1123025

Haidar Mokded - 1122383

Haidar Hershe – 1122496

**Phase 1**

**SDLC Model Chosen:**

**Agile Model**

For the development of the Movie Booking website, I have chosen the **Agile Software Development Life Cycle (SDLC) model**. The Agile approach is highly flexible and allows iterative development with continuous feedback, making it ideal for projects with evolving requirements and user needs, like a web application that may receive frequent updates and improvements.

**Reason for Choosing Agile:**

**Iterative Development**: Agile breaks the project into sprints, enabling incremental feature releases and quick adaptations to user needs.

**Customer Feedback Integration**: Regular feedback ensures the website evolves based on user expectations for a better experience.

**Faster Time to Market**: Functional components can be launched early, allowing movie bookings before full project completion.

**Team Collaboration**: Agile fosters teamwork among developers, designers, and managers to maintain alignment and efficiency.

**Feasibility Study for Movie Booking Website :**

**Technical Feasibility :**

**Technology**: The website will use established technologies like **HTML**, **CSS**, **JavaScript**, **Spring Boot**, and **Vue.js**. All required technologies are available and can be easily integrated.

**Expertise**: The team has the necessary skills for frontend and backend development.

**Conclusion**: The project is technically feasible with existing tools and expertise.

**Economic Feasibility :**

**Costs**: Development costs include labor, hosting, and maintenance. Operational costs include payment gateway fees and customer support.

**Revenue**: The platform will generate revenue through ticket sales and potential partnerships.

**Conclusion**: The project is economically viable, with potential returns covering costs.

**Market Feasibility :**

**Target Market**: Moviegoers preferring online bookings, especially in a post-pandemic world.

**Competition**: Competitors like Fandango exist, but differentiation through a better UI and exclusive features can help.

**Conclusion**: The market demand is high, and with proper differentiation, the platform has good market potential.

**Legal Feasibility :**

**Compliance**: The platform must follow data privacy regulations (GDPR, CCPA) and ensure payment security (PCI-DSS).

**Licensing**: Agreements with cinemas and payment providers are necessary.

**Conclusion**: Legally feasible with the right agreements and compliance with regulations.

**Operational Feasibility :**

**Staffing**: A small team will handle customer support and system maintenance.

**Integration**: Seamless integration with cinema systems and payment gateways.

**Conclusion**: The system is easy to operate and integrate with existing infrastructure.

**Scheduling Feasibility :**

**Timeline**: The project can be completed within 12 weeks—2 weeks for planning, 8 weeks for development, 2 weeks for testing, and 1 week for deployment.

**Conclusion:** The project can be completed on schedule.

**Conclusion :**

The Movie Booking Website is technically, economically, legally, operationally, and schedule-wise feasible. The project has a high chance of success if well-executed.

Revision history:

Version: 1.0

Author: Unknown

Version description: Initial version of the Software Requirements Specification document for the Movie Booking website.

Date completed: 21/3/2025

Review History:

Approving party: Dr. Ibrahim Dhaine

Version approved: 1.0

Signature:

Date:

Approval History:

Reviewer: Dr. Ibrahim Dhaine  
Version reviewed: 1.0

Signature:  
Date:

Table of Contents

1. Introduction  
   1.1 Product Scope  
   1.2 Product Value  
   1.3 Intended Audience  
   1.4 Intended Use  
   1.5 General Description
2. Functional Requirements
3. External Interface Requirements  
   3.1 User Interface Requirements  
   3.2 Hardware Interface Requirements  
   3.3 Software Interface Requirements  
   3.4 Communication Interface Requirements
4. Non-Functional Requirements  
   4.1 Security  
   4.2 Capacity  
   4.3 Compatibility  
   4.4 Reliability  
   4.5 Scalability  
   4.6 Maintainability  
   4.7 Usability  
   4.8 Other Non-Functional Requirements
5. Definitions and Acronyms

**1. Introduction**

The purpose of this Software Requirements Specification (SRS) document is to provide a detailed description of the functional and non-functional requirements for the Movie Booking website. This document is intended for use by the development team, project managers, and stakeholders to ensure a clear, shared understanding of the system's objectives, features, and constraints, guiding the design, development, and testing processes.

**1.1 Product Scope**

The product scope of this movie booking website is to offer users a convenient and seamless platform to browse movies, select showtimes, and reserve seats at cinemas. Users can view real-time availability, choose their preferred screen, and securely pay for tickets, receiving an e-ticket upon successful booking. The website also features a feedback system where users can read and share reviews, aiding in informed decision-making. Benefits include time-saving, ease of access, and a smooth booking process. Objectives focus on providing an intuitive interface, supporting multiple payment options, and ensuring real-time updates on availability, all aimed at enhancing user engagement, increasing conversion rates, and delivering a reliable, efficient booking experience to ensure high user satisfaction and repeat usage.

**1.2 Product Value**

The movie booking website offers convenience, efficiency, and a seamless user experience. Users can browse movies, check real-time showtimes, and book seats easily, avoiding queues. Secure payments and instant e-tickets ensure quick access, while feedback options enhance decision-making, making the booking process fast and reliable.

**1.3 Intended Audience**

The product is designed for moviegoers of all ages, offering a convenient way to browse, book, and pay for cinema tickets online.

**1.4 Intended Use**

The intended audience will use this product to easily browse available movies, view showtimes, and select their preferred seats at local cinemas. They will choose a movie, check real-time availability, and complete the booking process through a secure payment system. After booking, users will receive an e-ticket for entry. Additionally, users can read and share feedback to help them make informed decisions on their movie choices. The platform will be accessed via web or mobile devices for a quick, seamless movie booking experience.

**1.5 General Description**

The software will allow users to easily browse movies, check available showtimes, and select their preferred cinema and screen. Users can view real-time seat availability, book tickets, and pay securely online. After booking, users will receive an e-ticket for entry. The platform will also feature a feedback section where users can read and leave reviews. Additionally, users can create accounts to manage their bookings. The software will be accessible on both web and mobile devices for a smooth and convenient booking experience.

**2. Functional Requirements**

The movie booking website features a beautifully designed, simple, and intuitive user interface that ensures easy navigation across desktops, tablets, and mobile devices. The booking flow is streamlined, guiding users seamlessly from movie selection to payment and e-ticket generation. High-quality movie posters, clear icons, and buttons, along with a visually distinct feedback section, enhance user interaction and make the platform highly engaging. The website is compatible with major operating systems (Windows, macOS, Android, and iOS) and supports modern web browsers like Chrome, Firefox, Safari, and Edge. It integrates secure payment gateways such as Stripe or PayPal for smooth and reliable transactions. Real-time seat availability is accurately updated, and the system is designed to scale, handling high user volumes during peak times without performance issues. Additionally, the website provides a fully functional mobile experience, making it easy for users to browse, book, and pay for tickets on their smartphones and tablets.

**3. External Interface Requirements**

**3.1 User Interface Requirements**

The user interface of the movie booking website is designed to provide an intuitive and engaging experience, with clear, easy-to-navigate sections. The homepage showcases the latest movie trailers, online movies, popular films, and new releases, giving users immediate access to the most relevant content. The "Movies" section features popular movies, latest releases, categories, and trending films, making it easy for users to explore different genres and discover new favorites.

When users choose a movie, they can view the available cinema seats, with taken seats clearly marked. The seat selection process is simple: users pick their desired seats, view real-time availability, and proceed to payment. Once the payment is completed, users receive an e-ticket for their booking.

The "About" section provides information about the website, including key statistics such as the number of users, movies available, and staff members, along with user feedback to build trust and credibility. The "Contact" page allows users to send messages and easily get in touch with the support team for any inquiries or assistance.

The website’s style guide ensures a clean, modern design with consistent branding, readable fonts, contrasting colors, and intuitive icons. It is fully responsive, offering a seamless experience across desktops, tablets, and mobile devices. Tooltips and prompts guide users through the booking process, ensuring a smooth experience from browsing movies to receiving their tickets.

**3.2 Hardware Interface Requirements**

The movie booking website is designed to run on desktops, laptops (Windows/macOS), and mobile devices (iOS/Android), supporting modern browsers like Chrome, Firefox, Safari, and Edge. It requires a stable internet connection with at least 3 Mbps for browsing, seat selection, and payments, and 5 Mbps for streaming movie trailers. To ensure secure and efficient communication, the platform utilizes HTTPS for data security, WebSockets for real-time seat updates, TLS for secure payments, and SMTP/IMAP for sending e-ticket confirmations and notifications.

**3.3 Software Interface Requirements**

The movie booking website's frontend is built using HTML, CSS, and JavaScript, with a focus on creating a dynamic and responsive user interface. The site uses a frontend framework like Vue.js to handle interactions and manage state, ensuring a smooth experience for the user. The layout is styled with Bootstrap for consistent design and responsiveness across devices. The website communicates with external services through RESTful APIs to retrieve movie data, showtimes, seat availability, and to handle payment processing. Secure payment transactions are handled through integration PayPal. Additionally, WebSockets are implemented to provide real-time seat availability updates as users make selections

**3.4 Communication Interface Requirements**

The movie booking website allows users to easily contact the support team through an embedded contact form, where they can send inquiries or requests. When users enter their email address, they will receive updates and new suggestions, such as movie releases or special promotions, directly to their inbox. Additionally, booking confirmations, e-tickets, and payment updates are sent via email to keep users informed throughout their booking process.

**4. Non-Functional Requirements**

**4.1 Security**

The movie booking website follows strict privacy and data protection regulations, including GDPR for user data protection and PCI-DSS for secure payment processing. All user information is encrypted in transit and at rest, and secure HTTPS connections are used for all communications. The platform ensures compliance with relevant data protection laws, regularly conducting security audits to maintain a high level of security.

**4.2 Capacity**

The movie booking website currently stores user data, movie information, bookings, payment records, and e-tickets. As the site grows, storage needs will increase for a larger user base, more movies, and higher-quality media. Scalable cloud storage or dedicated servers will ensure fast access to data and maintain performance during high traffic.

**4.3 Compatibility**

The movie booking website requires a dual-core processor (Intel i3 or equivalent), 4 GB RAM, 500 MB storage, and a 1366 x 768 resolution for desktops, with a 3 Mbps internet connection. For mobile devices, it supports iOS 10.0+ or Android 6.0+, a quad-core processor (1.5 GHz), 2 GB RAM, 100 MB storage, and a 3 Mbps data/Wi-Fi connection for optimal performance.

**4.4 Reliability**

The critical failure time under normal usage would be approximately 5 minutes. This is the maximum acceptable downtime for key functions, such as seat selection, payment processing, and e-ticket generation, before it significantly impacts the user experience. If any critical features fail, such as showing available seats or completing payments, the system should restore them as quickly as possible, ideally within 5 minutes. This will ensure minimal disruption and prevent potential booking losses or customer dissatisfaction. Regular monitoring and quick response mechanisms will help maintain this reliability.

**4.5 Scalability**

The movie booking website can handle up to 5,000 simultaneous users browsing and booking tickets without performance issues. It processes up to 100 bookings per minute and supports 500 simultaneous seat selections across multiple showtimes. The payment system can handle up to 50 transactions per minute without any slowdowns or errors.

**4.6 Maintainability**

Continuous integration (CI) for your movie booking website should involve automatically running tests and building the project every time code changes are pushed to the repository. Tools like Jenkins, GitHub Actions, or CircleCI can be used to ensure that features and bug fixes are deployed quickly and reliably. After passing automated tests, code is deployed to staging for further testing before being pushed to production. This process ensures fast updates, minimizes downtime, and maintains website stability.

**4.7 Usability**

The movie booking website is designed to be highly user-friendly, with a clean and intuitive interface that allows users to easily browse movies, view available showtimes, and book seats. Navigation is straightforward, with clear sections for "Movies," "Shows," and "Book a Seat." The seat selection process is simple, showing real-time updates on available seats, and the payment process is quick and easy with minimal steps to complete the booking. The website is fully responsive, providing a smooth experience across both desktop and mobile devices. Users receive clear feedback, such as booking confirmations and error messages, ensuring a seamless and enjoyable experience.

**4.8 Other Non-Functional Requirements**

The movie booking website is designed for optimal performance and reliability, loading within 3 seconds and maintaining 99.9% uptime to provide a fast, responsive user experience. Data is regularly backed up, with a recovery time of 30 minutes in case of failure. The site supports multiple languages and complies with WCAG 2.1 accessibility standards. It also adheres to GDPR and CCPA data privacy regulations, ensuring user data is protected. The website is scalable, handling increasing user traffic seamlessly as demand grows.

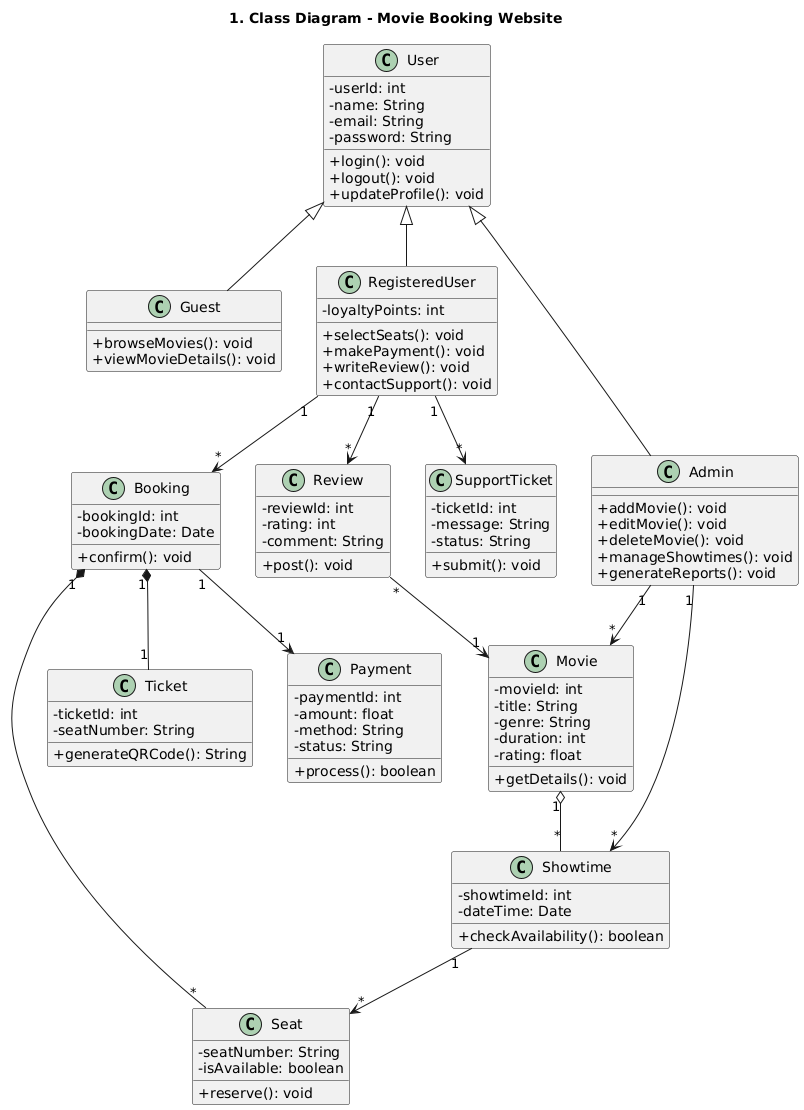
**5. Definitions and Acronyms**

1. E-ticket: A digital ticket issued to a user after booking, used for cinema entry.
2. Secure Payment Gateway: Technology that ensures the security of online payment transactions.
3. Frontend Framework: Tools and libraries for building the client-side interface and handling user interactions.
4. Backend Integration: Connecting the frontend with server-side databases and applications.
5. GDPR (General Data Protection Regulation): EU regulation that protects the personal data and privacy of citizens.
6. PCI-DSS (Payment Card Industry Data Security Standard): Security standards for safely handling credit card transactions.
7. HTTPS (HyperText Transfer Protocol Secure): Secure version of HTTP that encrypts data between user browsers and servers.
8. TLS (Transport Layer Security): A protocol ensuring secure communication over a network.
9. SMTP (Simple Mail Transfer Protocol): Protocol for sending emails between servers.
10. IMAP (Internet Message Access Protocol): Protocol for retrieving emails from a server.
11. WebSocket: A protocol for enabling real-time communication between a server and a client.
12. WCAG (Web Content Accessibility Guidelines): Guidelines to make web content more accessible to users with disabilities.

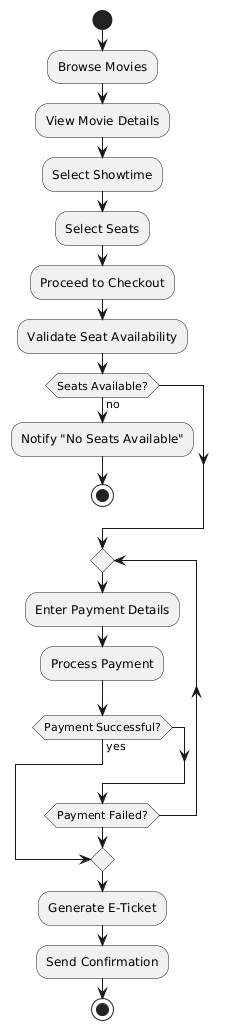
CI (Continuous Integration): A development practice where code changes are regularly integrated and tested automatically.

**Phase 2**

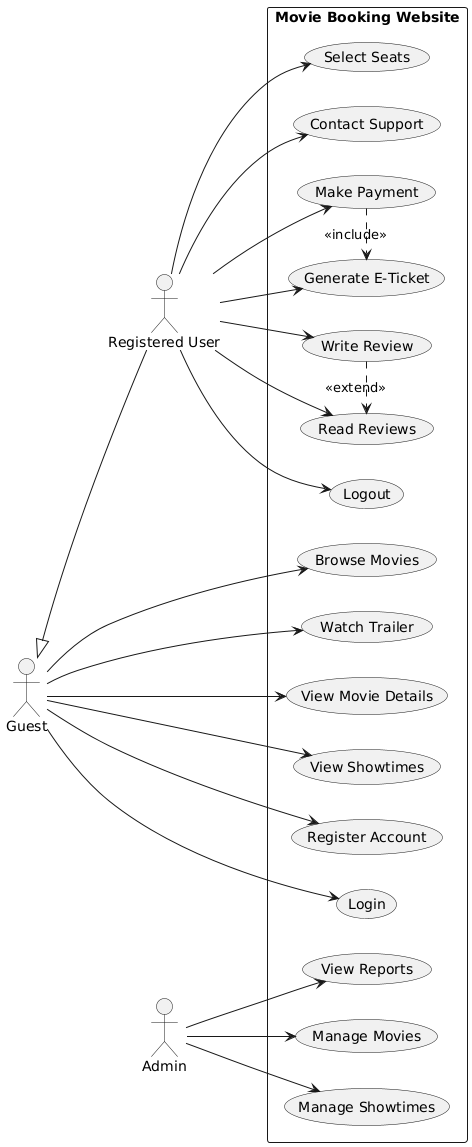
**1 - Class Diagram**

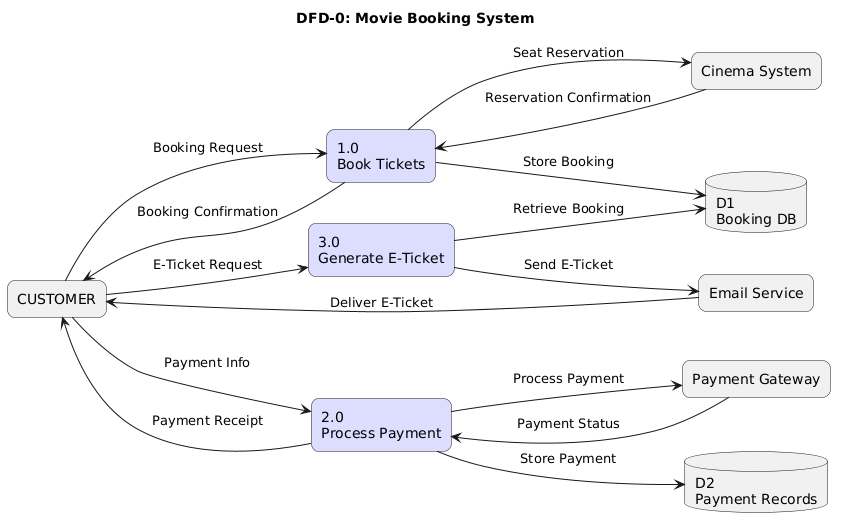
****

**2 - Activity Diagram**

****

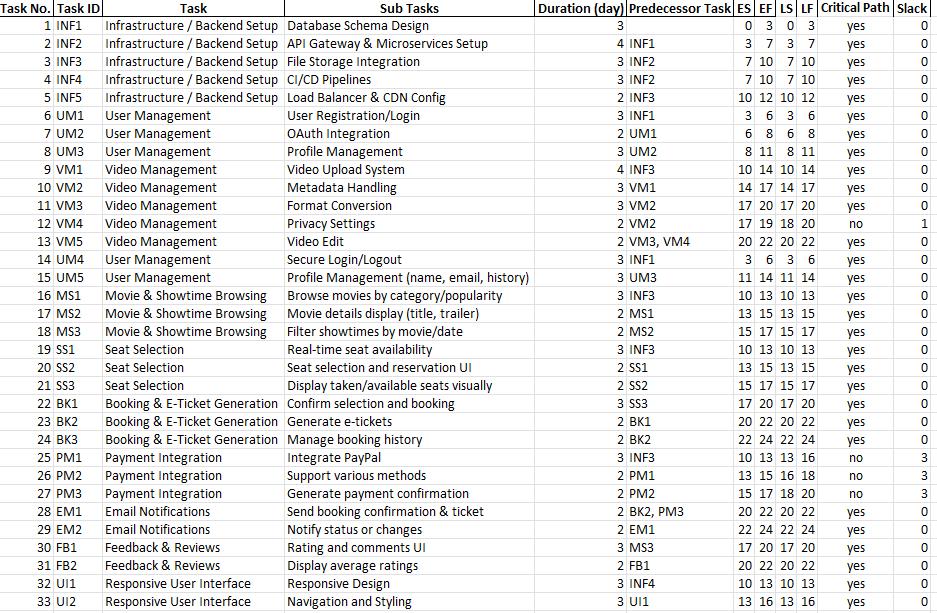
**3 - Use Case Diagram**

****

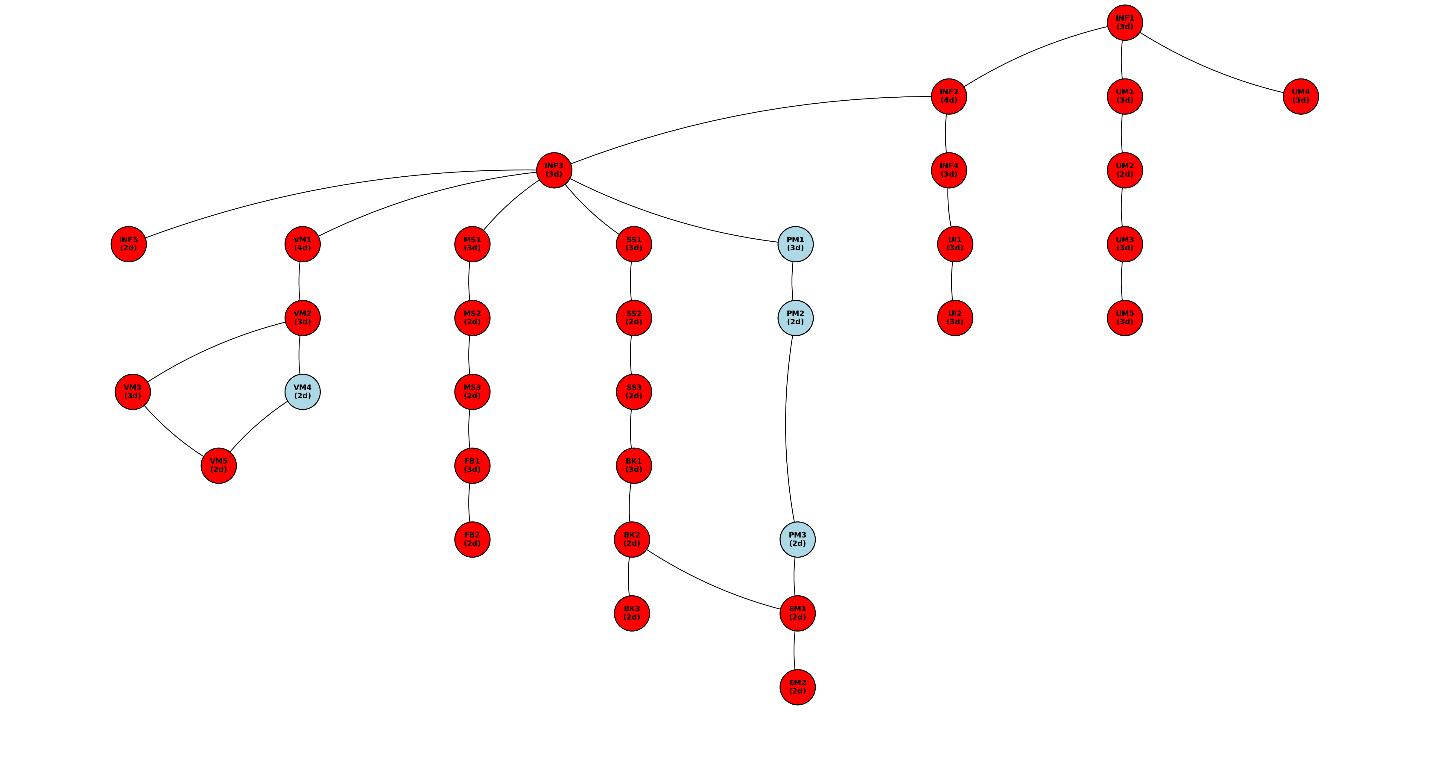
**4 - Data Flow Diagram** 

**Phase 3**

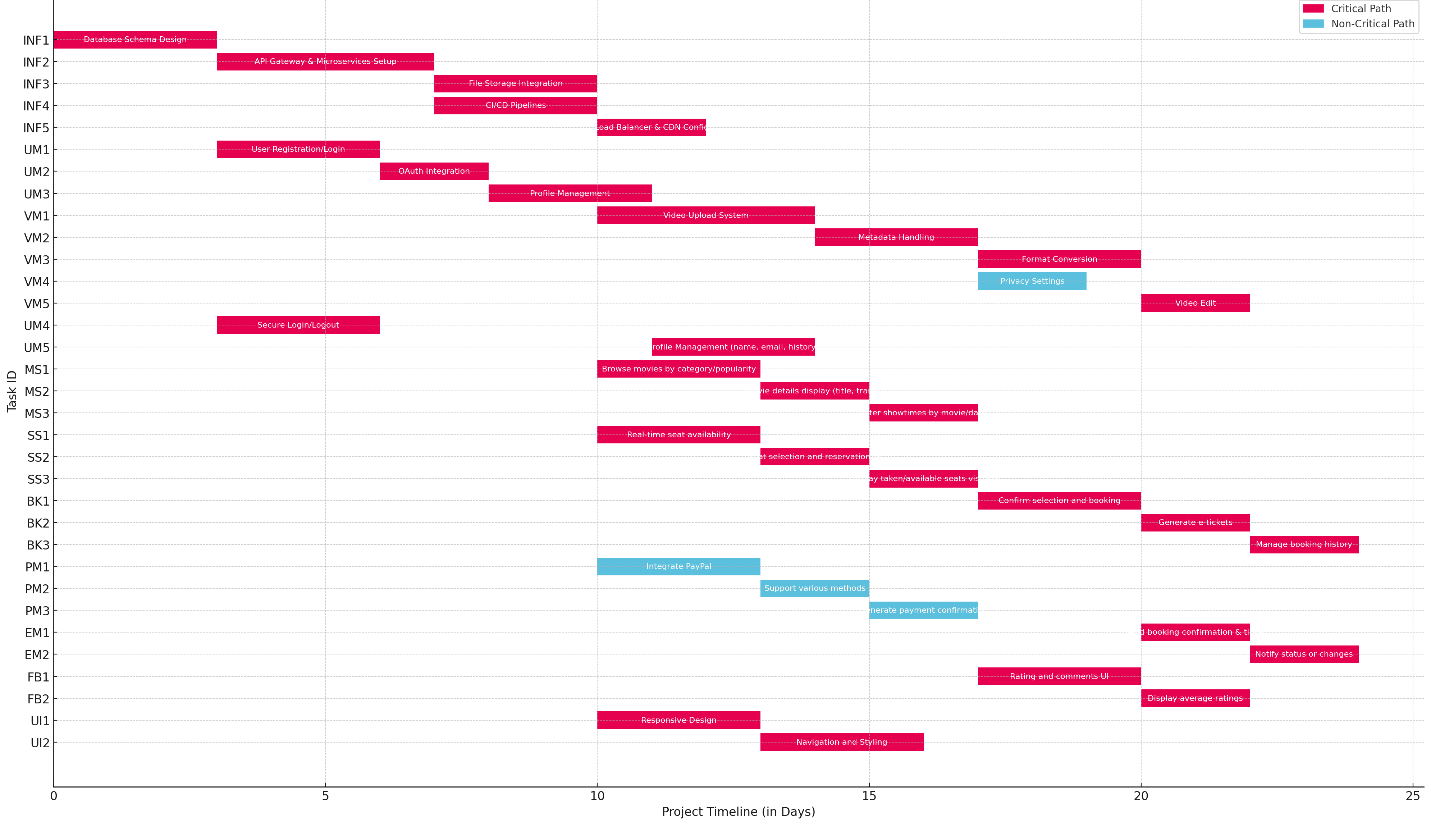
**1 - Table**

****

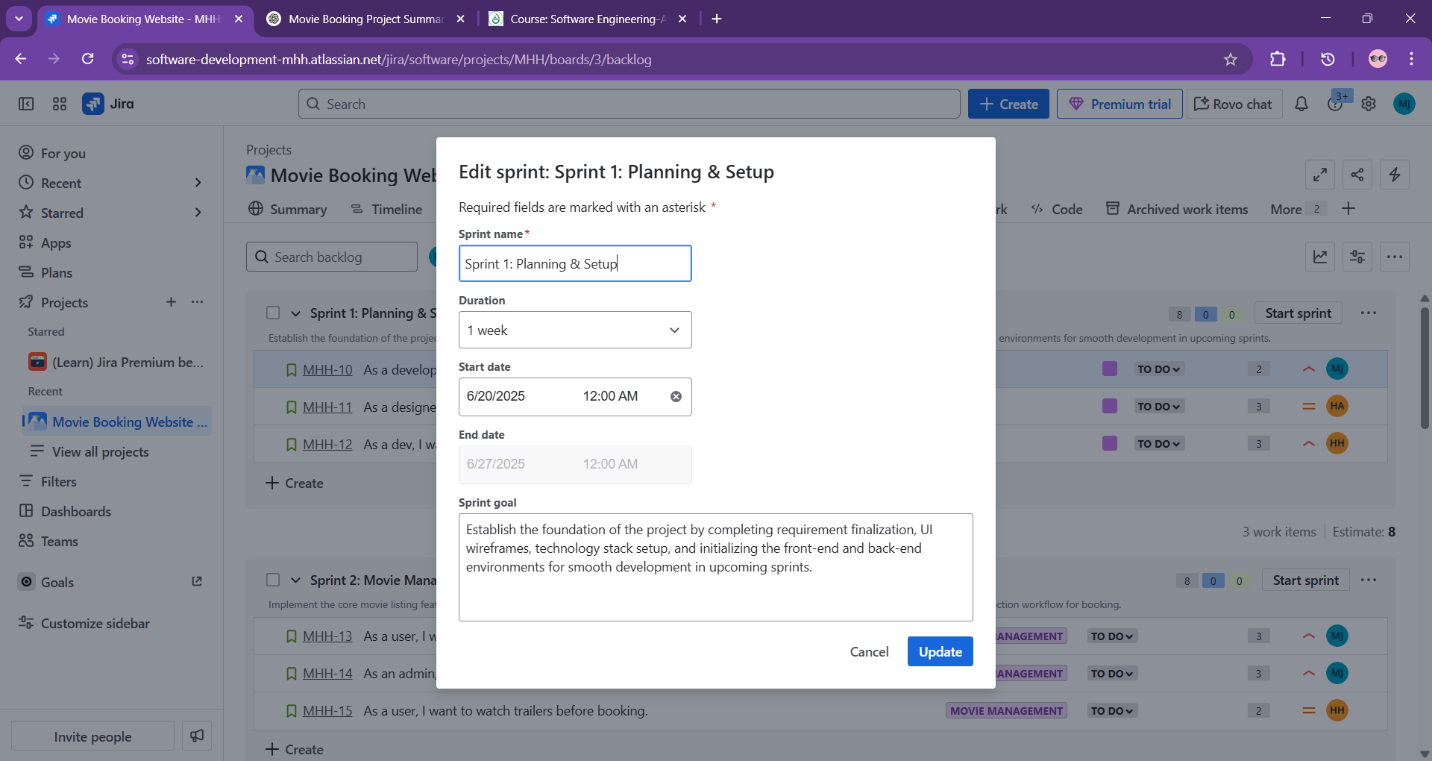
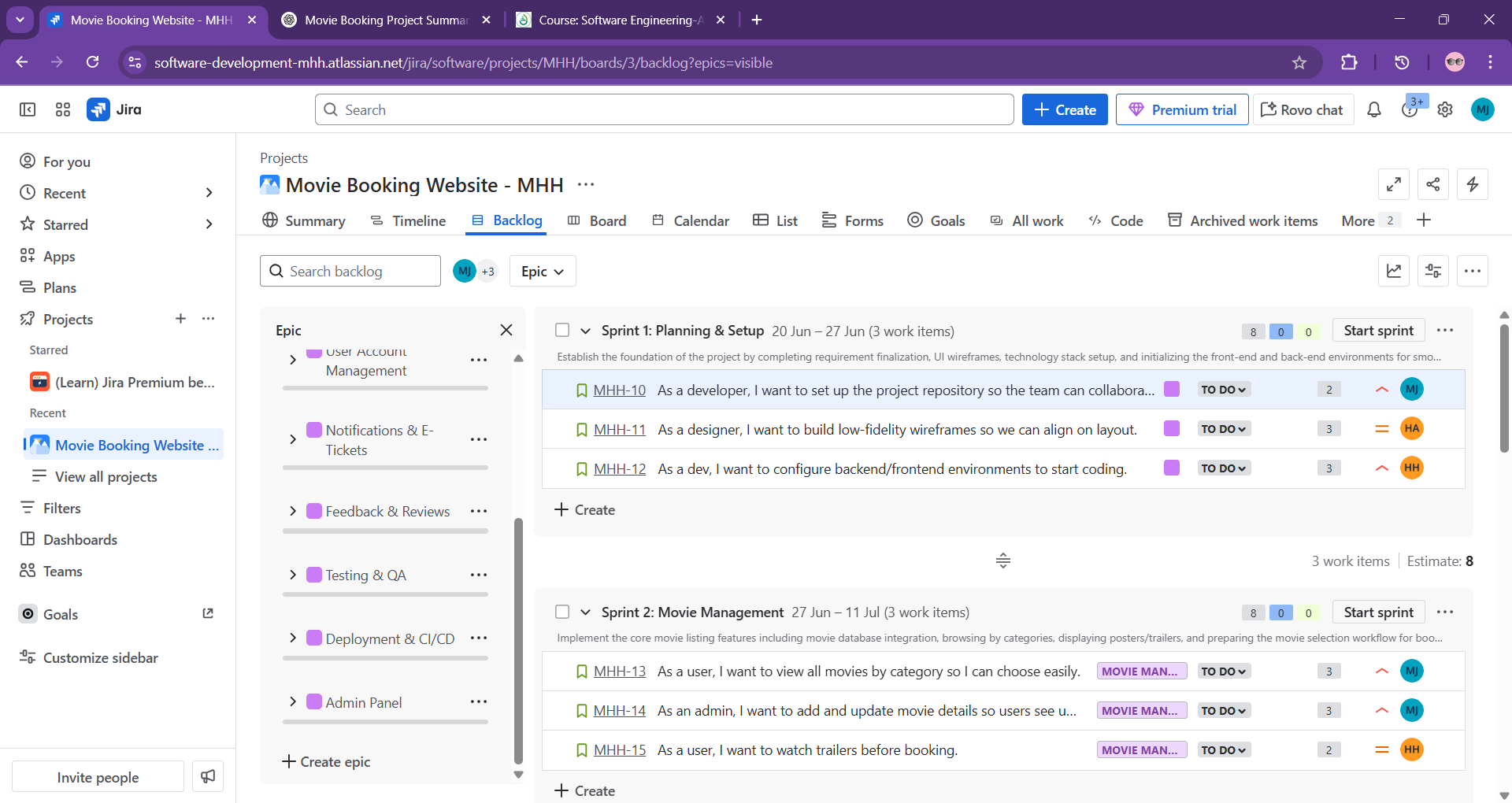
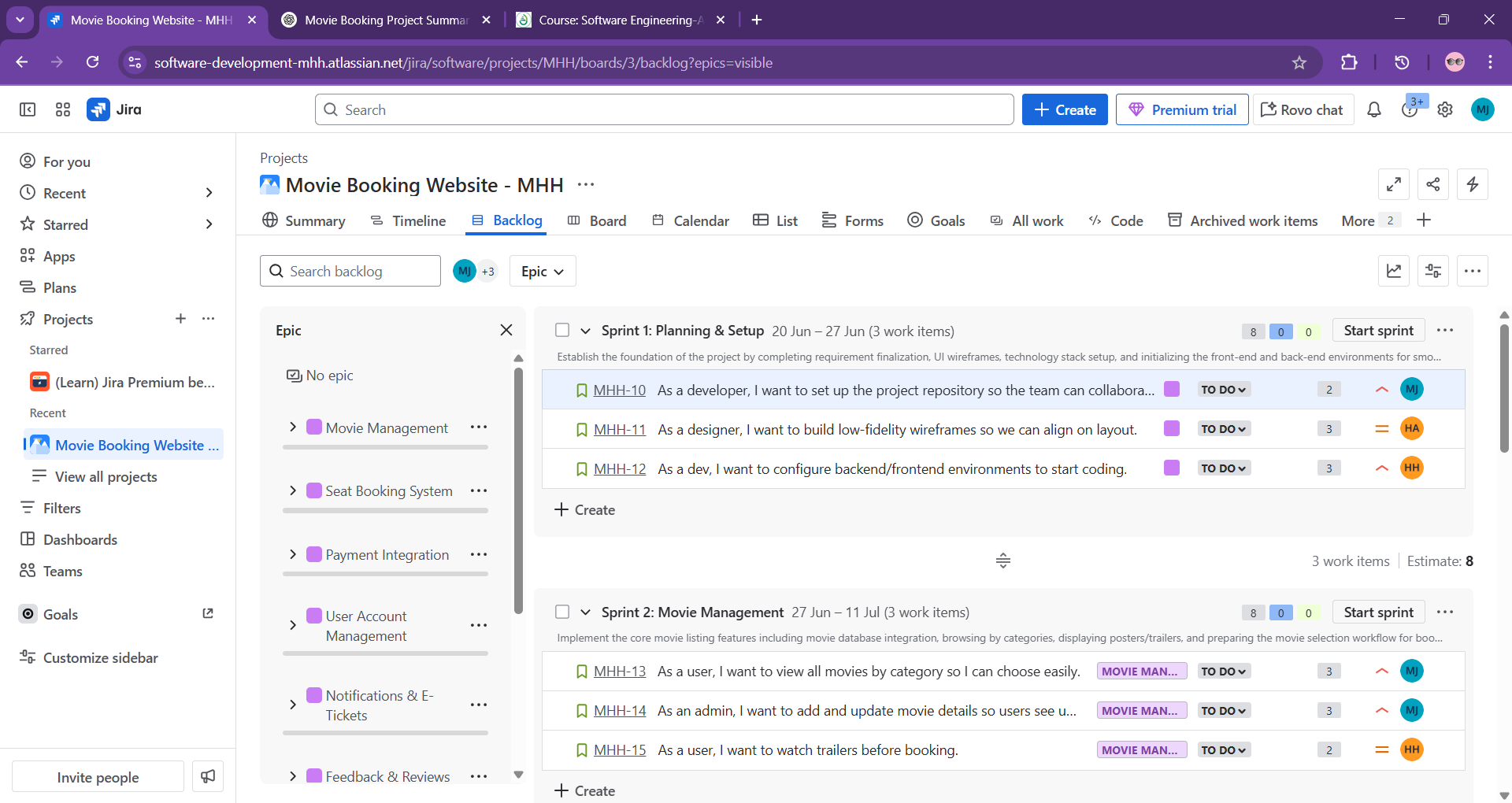
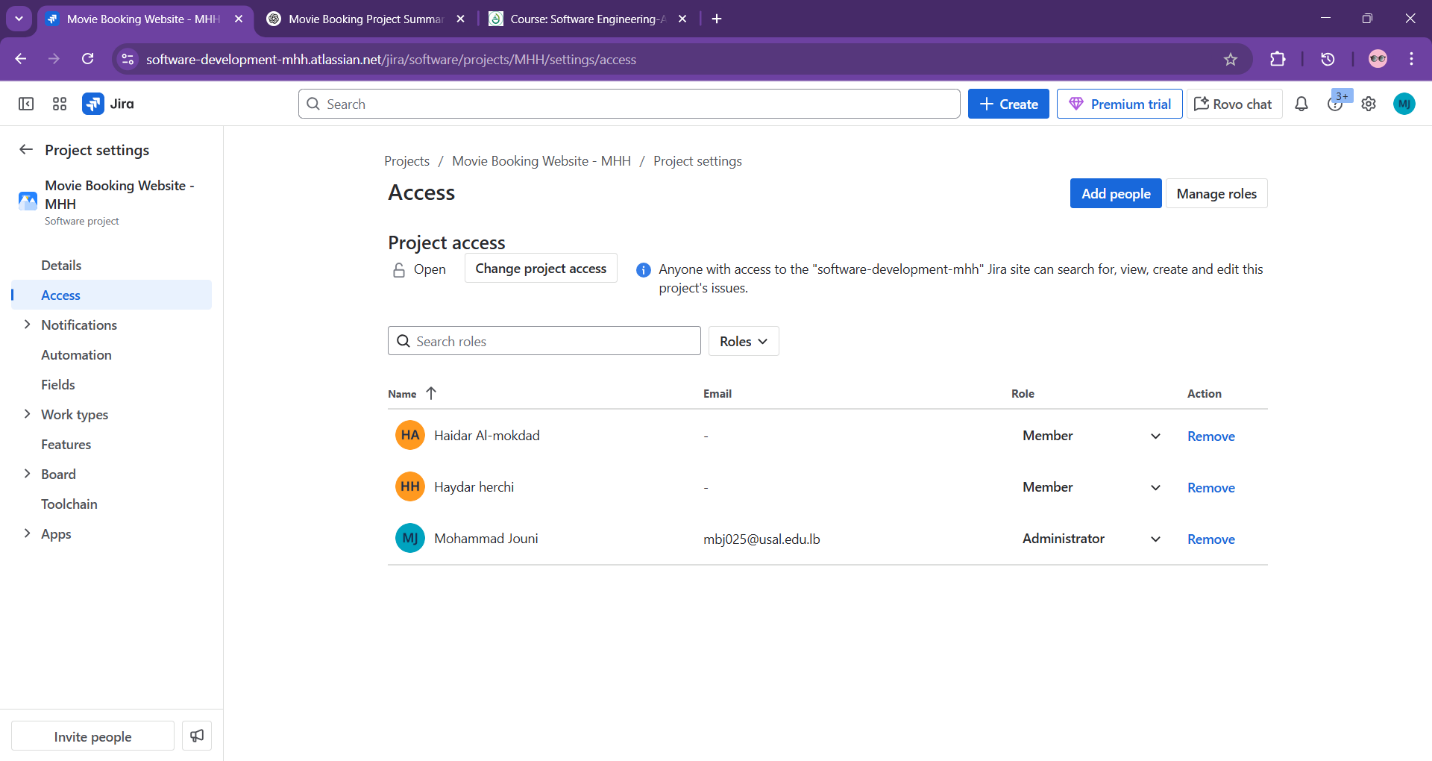
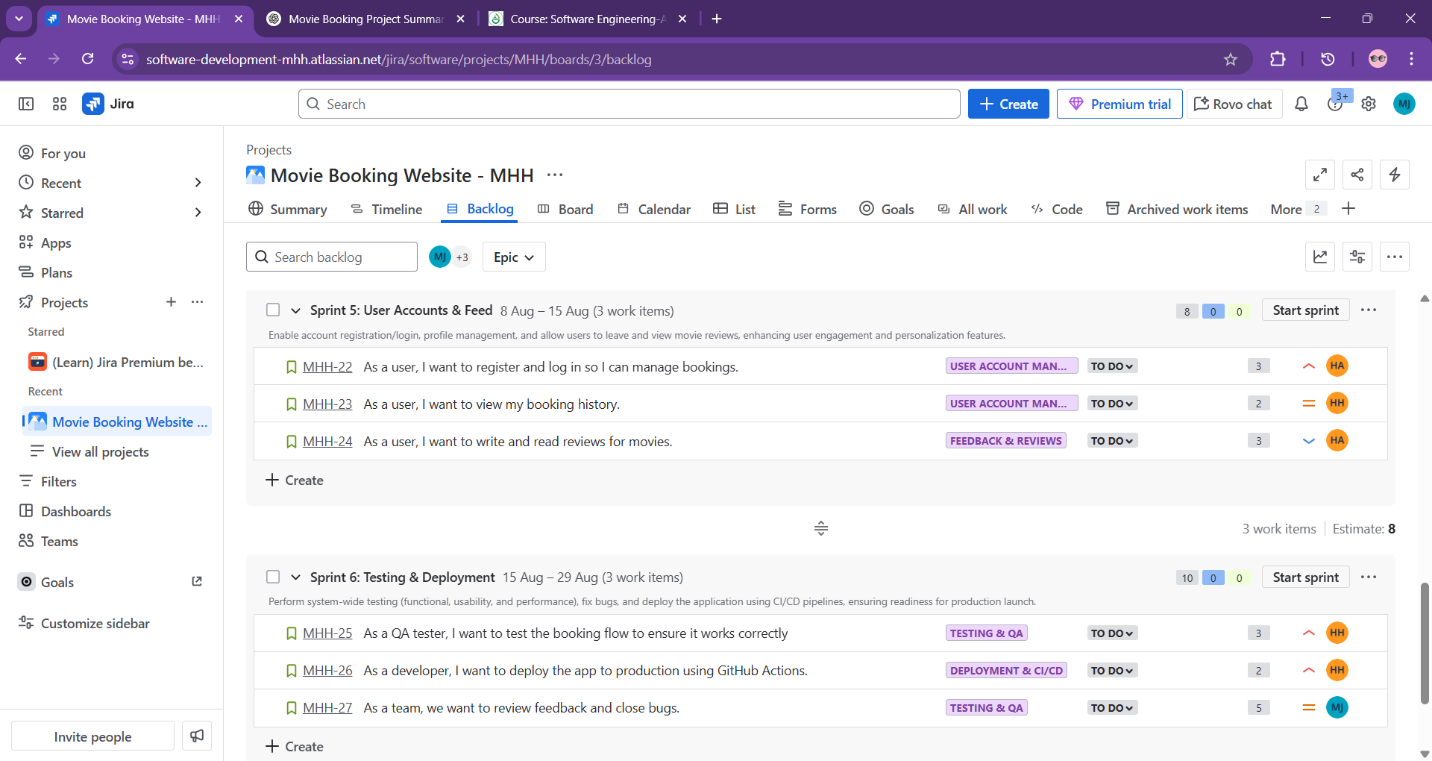
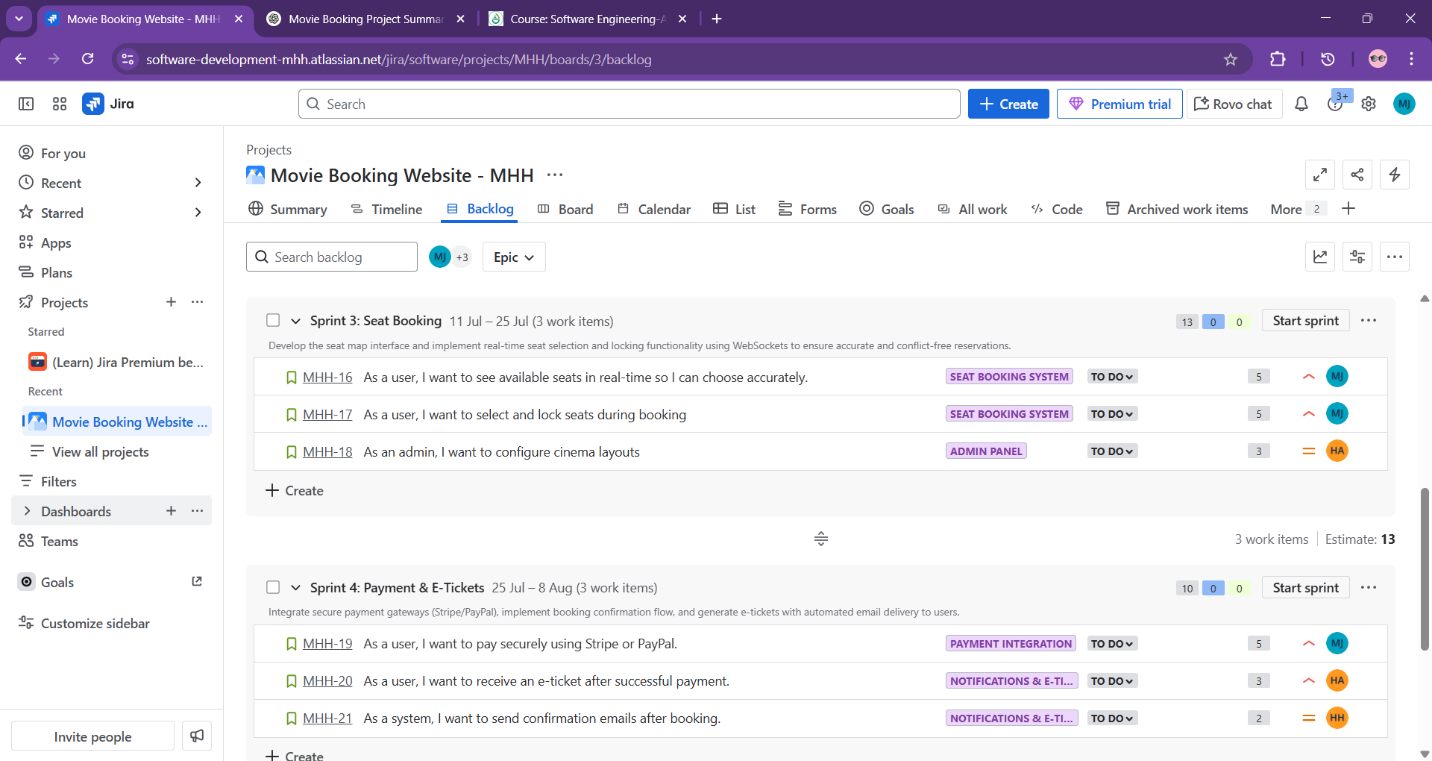
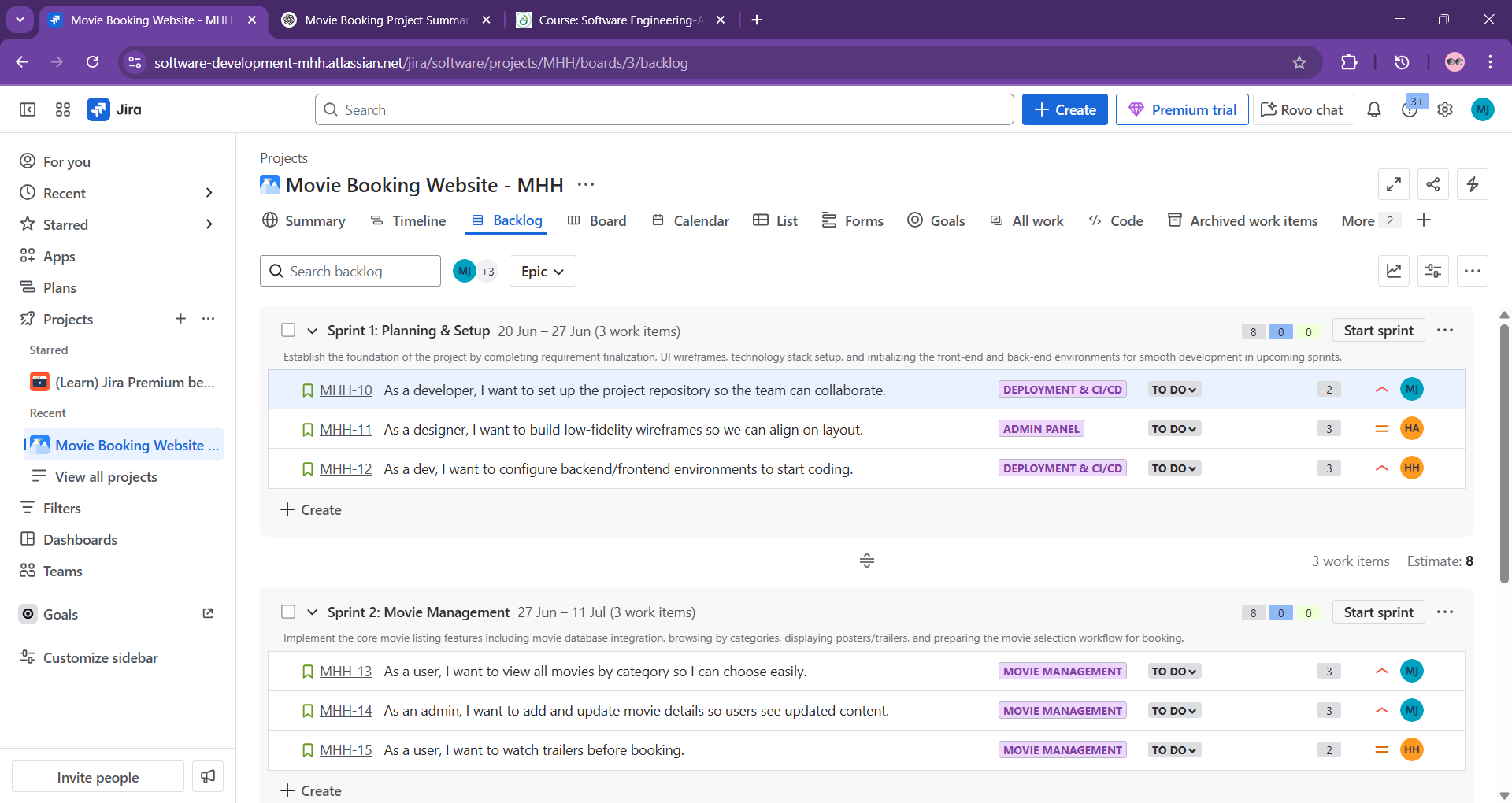
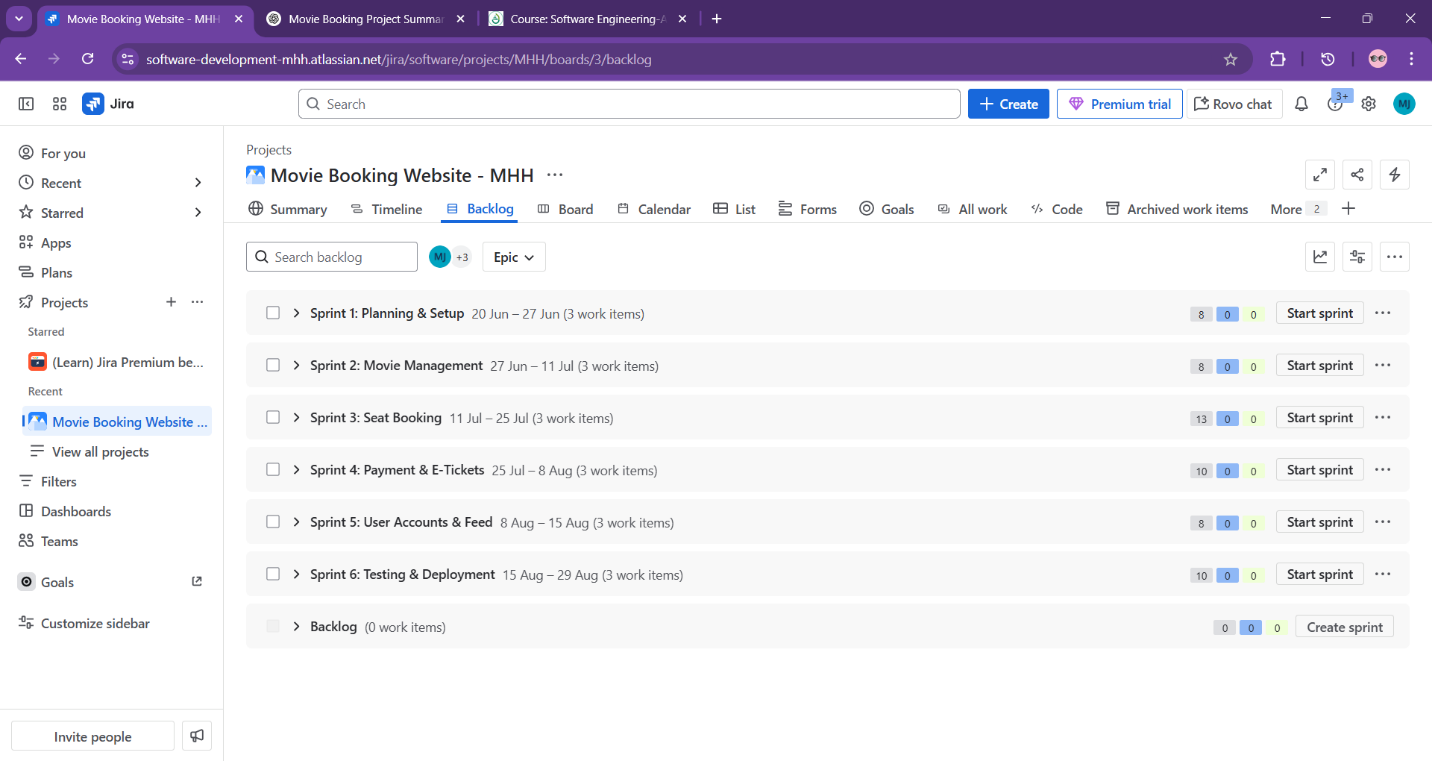
**2 - Pert Chart**

****

**3 - Gantt Chart**



**Phase 4**



**Phase 5**

