Chirayu Batra

21BCE5756

(This assignment is done with my team partner Jay H. Khania(21BCE1394))

Software Requirements Specification

for

MediNet

Version 1.0 approved

Prepared by Chirayu Batra(21BCE5756), Jay H. Khania(21BCE1394)

School Of Computer Science And Engineering

8 June, 2023

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Chirayu Batra	8 June,23	Additional Requirement from stakeholders	1.1
Jay H. Khania	8 June,23	resolving the Bug issues	1.2

1. Introduction

1.1 Purpose

The purpose of this SRS is to outline both the functional and non-functional requirements of the subject Online Medicine Delivery system. In addition to said requirements, the document also provides a detailed profile of the external interfaces, performance considerations and design constraints imposed on the subsequent implementation. The document should act as a foundation for efficient and well-managed project completion and further serve as an accurate reference in the future.

1.2 Document Conventions

OMOS- Online Medicine Ordering System
SRS - Software Requirements Specification
DBMS Database Management System
LAN Local Area Network
IP Internet Protocol
TCP Transmission Control Protocol
UDP User Datagram Protocol
IEEE 802.11 Wireless Local Area Network Standard

1.3 Intended Audience and Reading Suggestions

The primary audience of this SRS document will be the development team employed to implement the specified Online Medicine Delivery system. It will not only provide an extensive capacity for project planning and progress assessment but it will further assist with stakeholder interactions. The secondary document audience comprises the stakeholders of the project, that is, pharmacists and associated staff. To this audience group, this SRS should convey and confirm the required functionality and represent a contractual agreement between the involved parties.

1.4 Product Scope

In current physical medicine stores, some form of physical static display is utilized to convey the available supplement choices to customers. Said menus are generally paper based and hence impose restrictions on the textual real estate available and the ability a medical store has to update them. The related concepts are encompassed by the general scope of the Online Medicine Ordering System. It is to replace display based and non-convenient forms of buying medicine using an electronic format.

1.5 References

C. Larman, APPLYING UML AND PATTERNS An Introduction to Object-Oriented
Analysis and Design and Iterative Development, 3rd ed., Massachusetts: Pearson Education, 2005.
D. Carrington, CSSE3002 Course Notes, School of ITEE University of Queensland, 2008
IEEE Recommended Practice for Software Requirements Specifications, IEEE Standard 830, 1998.

2. Overall Description

2.1 Product Perspective

The software described in this SRS is the software for a complete Online medicine ordering system. The system merges various hardware and software elements and further interfaces with external systems, it relies on a number of external interfaces for persistence and unhandled tasks,as well as physically interfacing with humans.

2.2 Product Functions

The Online Medicine Delivery ordering system interfaces with an existing payment system, including cash register and software accessible credit system, in order to quickly and easily handle customer billing. The payment system should be operable such that it can return information to the OMOS system as to whether payment was successful or failed.

2.3 User Classes and Characteristics

There are three separate user interfaces used by the OMOS software, each related to an interfaced physical hardware device . These three user interfaces are the Surface Computer UI ,Tablet UI and Display UI.

2.4 Operating Environment

The Surface Computer UI is the interface used by medicine customers. This interface uses the surface computer paradigm - users interact with the system by dragging 'objects' around on the flat screen touch-sensitive display.The Tablet UI is designed to run on a small, wireless-enabled touch-screen tablet PC, to be used by pharmacists to accommodate customer needs.The Display UI provides department wise staff with simple functionality related to ordered medicines.

2.5 Design and Implementation Constraints

The OMOS should be written in an object-oriented language with strong GUI links and a simple, accessible network API. The primary candidate tool chains are Java/Swing, C++/Qt and Python/Qt. The system must provide a capacity for parallel operation and system design should not introduce scalability issues with regard to the number of surface computers, tablets or displays connected at any one time. The system must be reliable enough to run crash and glitch free more or less indefinitely, or facilitate error recovery strong enough such that glitches are never revealed to its end-users.

2.6 User Documentation

The end-users of the OMOS fall into three primary categories, unskilled, partly skilled and highly skilled.

2.7 Assumptions and Dependencies

The SRS assumes that none of the constituent system components will be implemented as embedded applications.It is further assumed that tablet PCs of sufficient processing capability and battery life will be utilized.

3. External Interface Requirements

3.1 User Interfaces

This interface uses the surface computer paradigm - users interact with the system by dragging 'objects' around on the flat screen touch-sensitive display. For the OMOS, users can manipulate objects such as medicines, shampoo requirements, tips and menus on the surface of their table. Such objects can be moved into static objects such as medicine carts and payments to perform various functions. In addition to this object manipulation paradigm, a limited system menu is necessary. Users will summon their medicinal options, which is combined with a system/command menu, using an easy touch gesture, a double-tap on the touch surface, and dismiss it with a similar gesture or by tapping a close button GUI element.

3.2 Hardware Interfaces

These devices are the surface computers, the wireless tablets and the touch displays. All three devices must be physically robust and immune to liquid damage and stains. The devices(with the possible exception of displays) must also have good industrial design aesthetics, as they are to be used in place of normal medicine stores and notepads and will be in direct contact with customers.

3.3 Software Interfaces

The OMOS will interface with a Database Management System (DBMS) that stores the information necessary for the RMOS to operate. The DBMS must be able to provide, on requestand with low latency, data concerning the list of medicines available, employees (and their passwords) and available company wise medicines for the same disease or symptom.

3.4 Communications Interfaces

The OMOS will interface with a Local Area Network (LAN) to maintain communication with all its devices. It should use a reliable-type IP protocol such as TCP/IP or reliable-UDP/IP for maximum compatibility and stability. All devices it will interface with should contain standard Ethernet compatible, software accessible LAN cards to maintain communication between the serverand the surface computers, tablets, displays and the external payment system.

4. Other Nonfunctional Requirements

This subsection presents the identified non-functional requirements for the subject OMOS.The subcategories of non-functional requirements given are performance, safety, security requirements.

4.1 Performance Requirements

The server shall be capable of supporting an arbitrary number of surface computers, tablets and displays, that is, it shall provide no limit on how many devices are in the system.The server shall be capable of supporting an arbitrary number of active customer payments, that is, no payments shall be lost under any circumstances.

4.2 Safety Requirements

The system shall log every state and state change of every surface computer, tablet and display to provision recovery from system failure.The system shall be capable of restoring itself to its previous state in the event of failure(e.g. a system crash or power loss).The system shall be able to display a menu at all times to facilitate manual order taking should the need arise.The system shall utilize periodic 30-second keep-alive messages between tablets and the server to monitor tablet operational status.

4.3 Security Requirements

The application is password protected and also any update of new product entries and order processing is done by only privileged users.

4.4 Software Quality Attributes

The software shall be capable of supporting an arbitrary number of surface computers,tablets and displays, that is, it shall provide no limit on how many devices are in the system. It shall be capable of supporting an arbitrary number of medicines, that is, no medicines from the cart shall be lost under any circumstances or should not be misdelivered at any circumstance.

4.5 Business Rules

Only the admin can add products in the database. The user can only search and buy for the products added by the admin.The admin cannot buy products.

5. Other Requirements

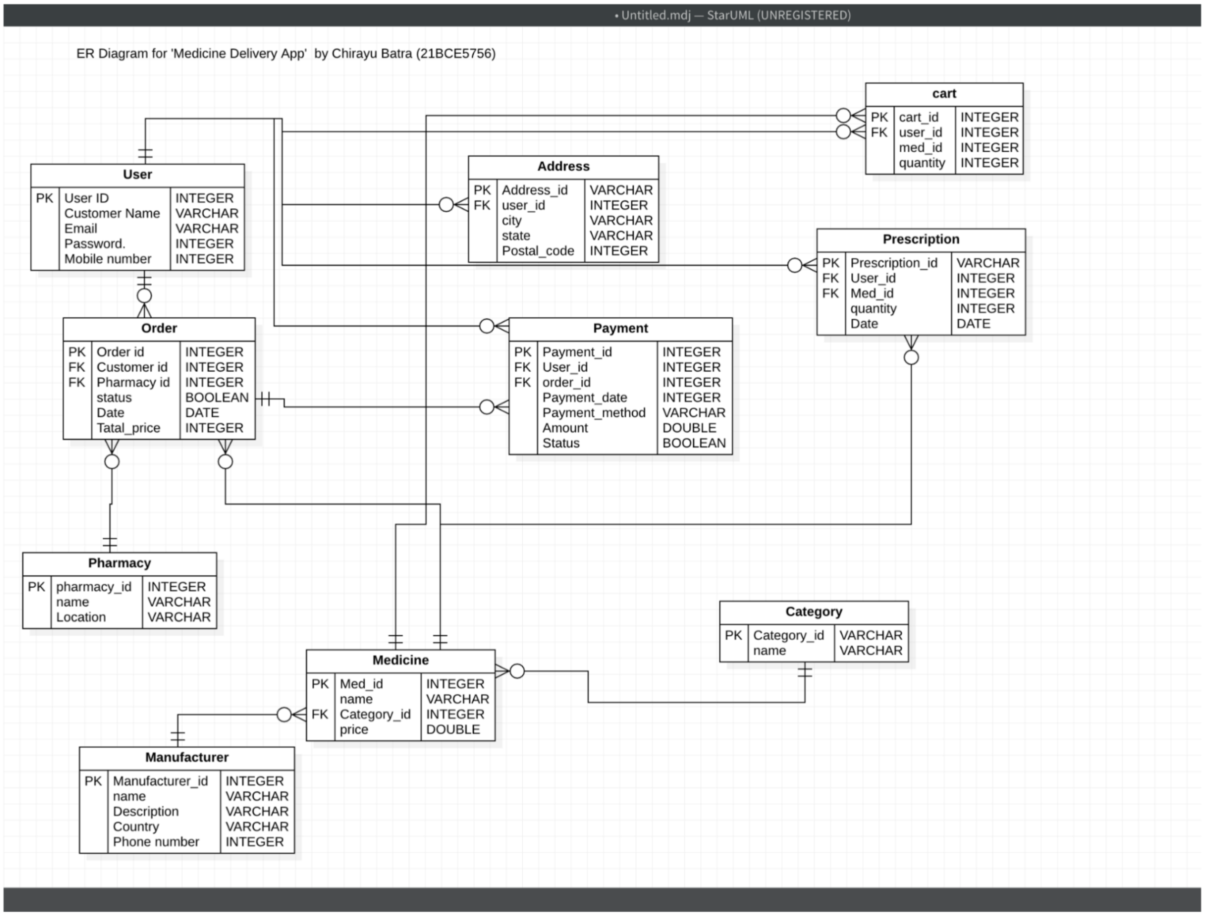
Appendix A: Glossary

HTTP:HyperText Transfer Protocol
SQL: Structured Query Language
GUI: Graphical User Interface SRS Software Requirements Specification
CSS: Cascading Style Sheet
PHP: a server-side scripting language designed for web development (Hypertext Preprocessor)

Appendix B: Analysis Models

ER diagram and DFD are given below:

ER diagram:



ER Diagram for 'Medicine Delivery App' by Chirayu Batra (21BCE5756)

cart

PK	cart_id	INTEGER
FK	user_id	INTEGER
	med_id	INTEGER
	quantity	INTEGER

Address

PK	Address_id	VARCHAR
FK	user_id	INTEGER
	city	VARCHAR
	state	VARCHAR
	Postal_code	INTEGER

User

PK	User ID	INTEGER
	Customer Name	VARCHAR
	Email	VARCHAR
	Password.	INTEGER
	Mobile number	INTEGER

Prescription

PK	Prescription_id	VARCHAR
FK	User_id	INTEGER
FK	Med_id	INTEGER
	quantity	INTEGER
	Date	DATE

Order

PK	Order id	INTEGER
FK	Customer id	INTEGER
FK	Pharmacy id	INTEGER
	status	BOOLEAN
	Date	DATE
	Tatal_price	INTEGER

Payment

PK	Payment_id	INTEGER
FK	User_id	INTEGER
FK	order_id	INTEGER
	Payment_date	INTEGER
	Payment_method	VARCHAR
	Amount	DOUBLE
	Status	BOOLEAN

Pharmacy

PK	pharmacy_id	INTEGER
	name	VARCHAR
	Location	VARCHAR

Category

PK	Category_id	VARCHAR
	name	VARCHAR

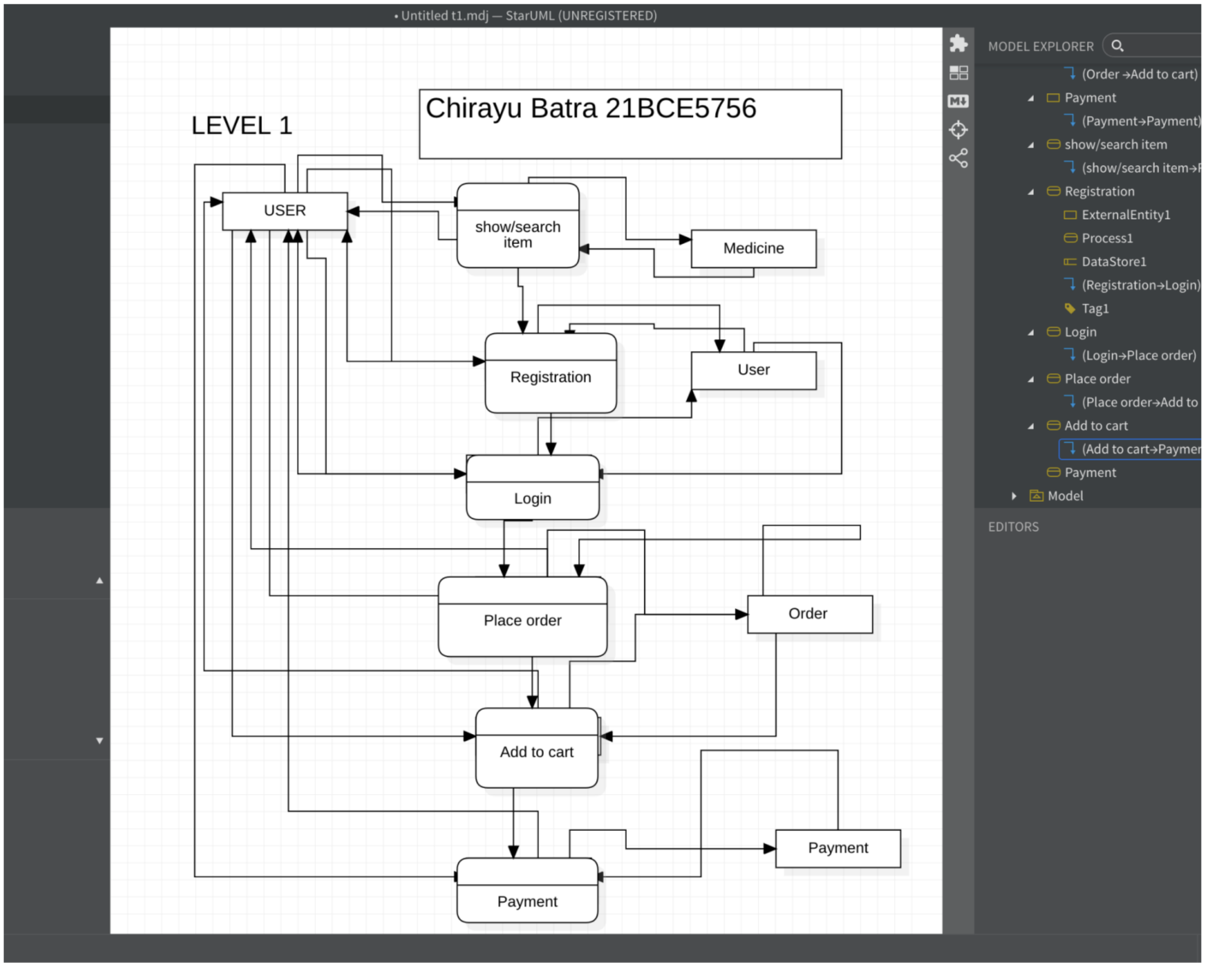
Medicine

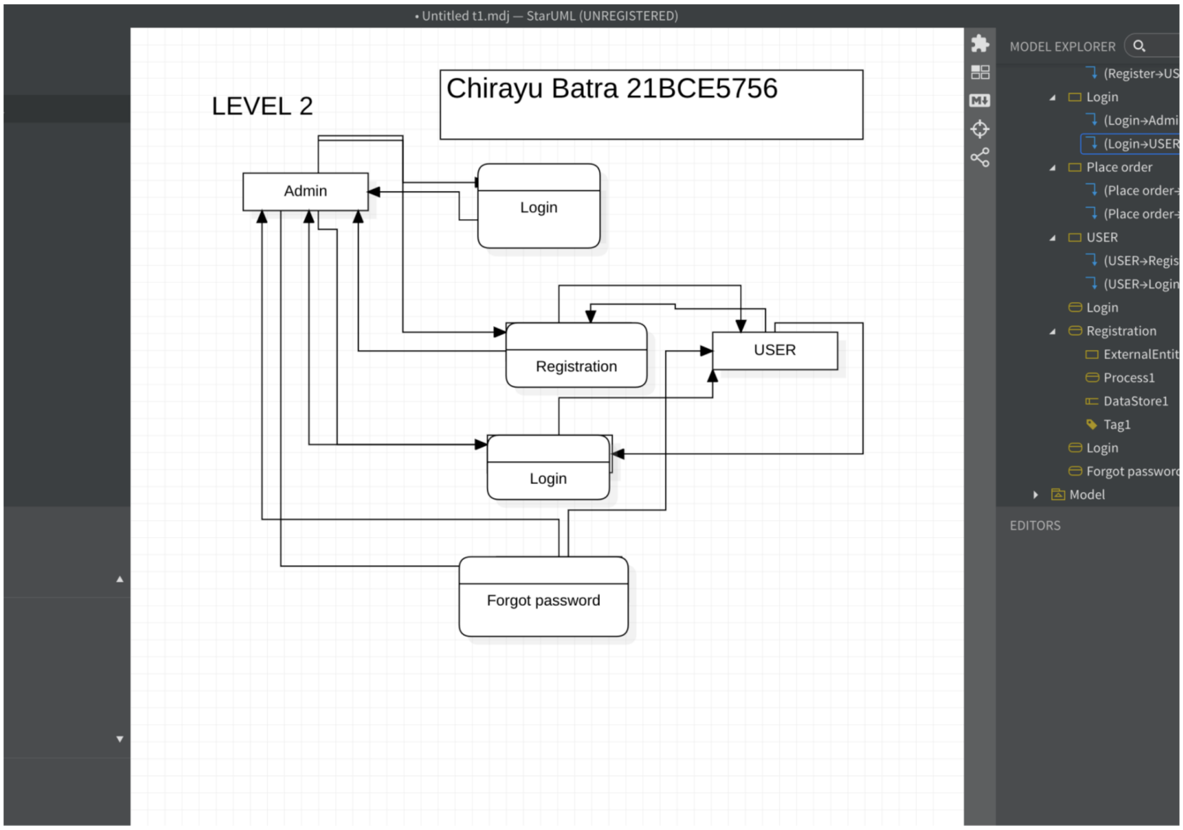
PK	Med_id	INTEGER
	name	VARCHAR
FK	Category_id	INTEGER
	price	DOUBLE

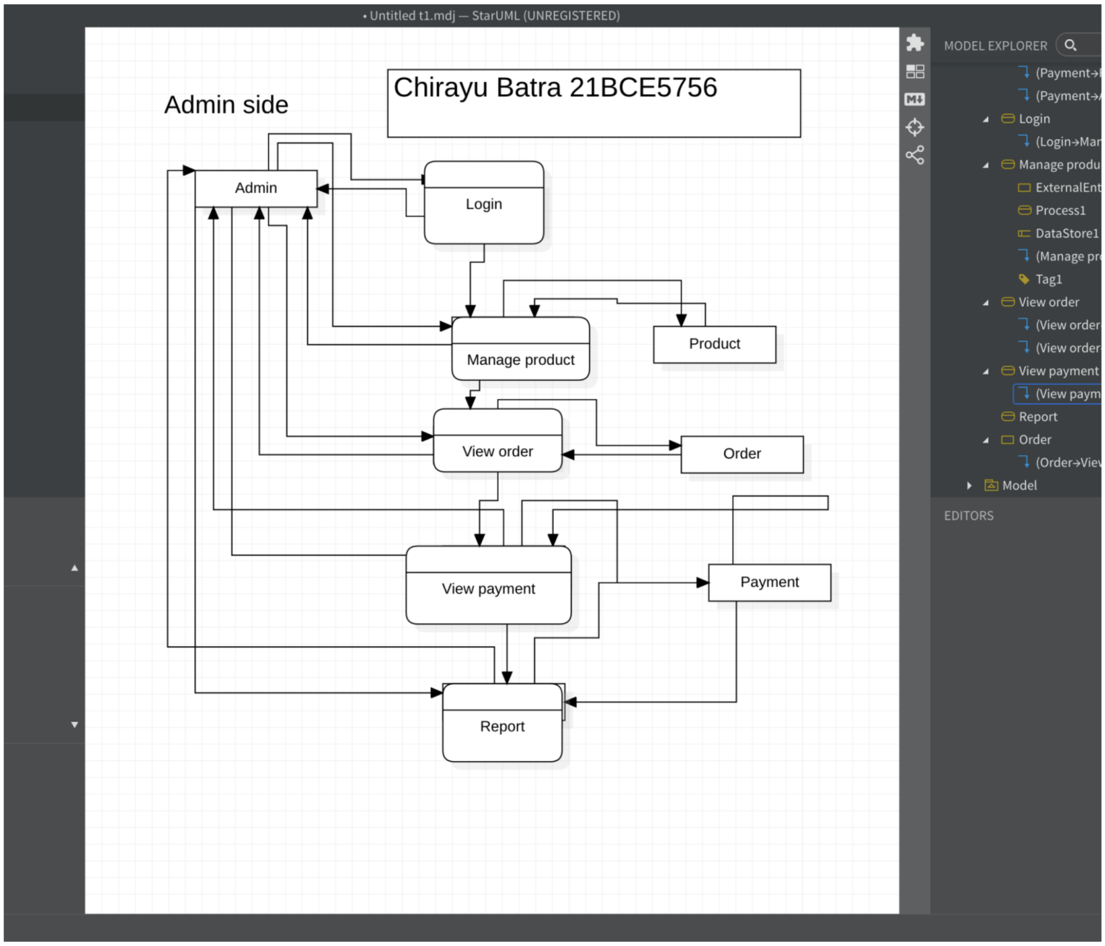
Manufacturer

PK	Manufacturer_id	INTEGER
	name	VARCHAR
	Description	VARCHAR
	Country	VARCHAR
	Phone number	INTEGER

DFD diagram:


• Untitled t1.mdj — StarUML (UNREGISTERED)
Chirayu Batra 21BCE5756
LEVEL 1
USER
show/search item
Medicine
Registration
User
Login
Place order
Order
Add to cart
Payment
Payment
MODEL EXPLORER
(Order →Add to cart)
Payment
(Payment→Payment)
show/search item
(show/search item→
Registration
ExternalEntity1
Process1
DataStore1
(Registration→Login)
Tag1
Login
(Login→Place order)
Place order
(Place order→Add to
Add to cart
(Add to cart→Paymer
Payment
Model
EDITORS


• Untitled t1.mdj — StarUML (UNREGISTERED)
LEVEL 2
Chirayu Batra 21BCE5756
Admin
Login
Registration
USER
Login
Forgot password
MODEL EXPLORER
(Register→US
Login
(Login→Admi
(Login→USER
Place order
(Place order→
(Place order→
USER
(USER→Regis
(USER→Login
Login
Registration
ExternalEntit
Process1
DataStore1
Tag1
Login
Forgot passwor
Model
EDITORS



Appendix C: To Be Determined List

- None

EXPERIMENT

TOPIC : AGILE STORY-BOARDING

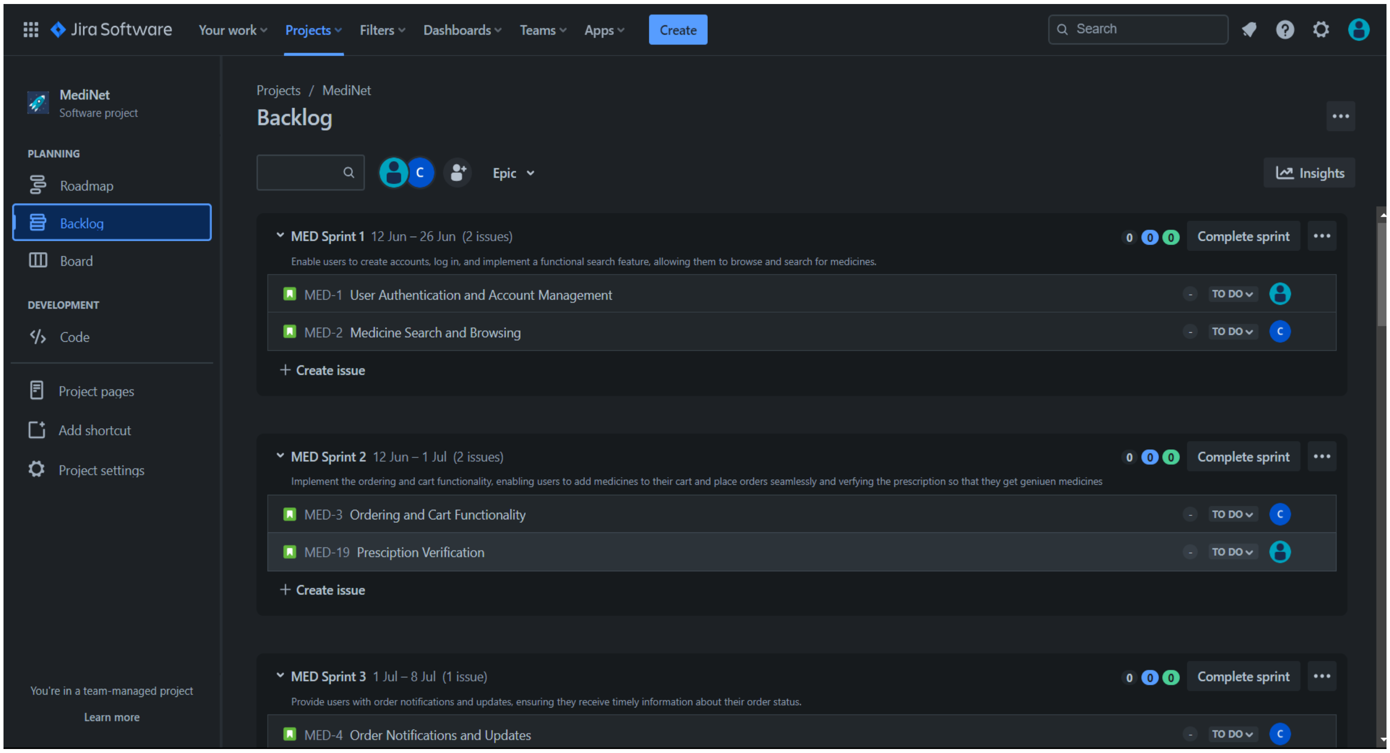
Aim:

To create user stories for "ONLINE MEDICINE DELIVERY APP" and demonstrate test cases for the same.

Description:

A user story is a short and simple description of a feature or a requirement from the perspective of the end user or customer. A scrum is a framework for managing complex projects by breaking them down into smaller chunks called epics, which are further divided into user stories. Scrum helps teams collaborate and deliver value to the customer in short iterations called sprints.

Output:



Backlog

MED Sprint 1 12 Jun – 26 Jun (2 issues) 0 0 0

Enable users to create accounts, log in, and implement a functional search feature, allowing them to browse and search for medicines.

MED-1	User Authentication and Account Management	TO DO	
MED-2	Medicine Search and Browsing	TO DO	C

MED Sprint 2 12 Jun – 1 Jul (2 issues) 0 0 0

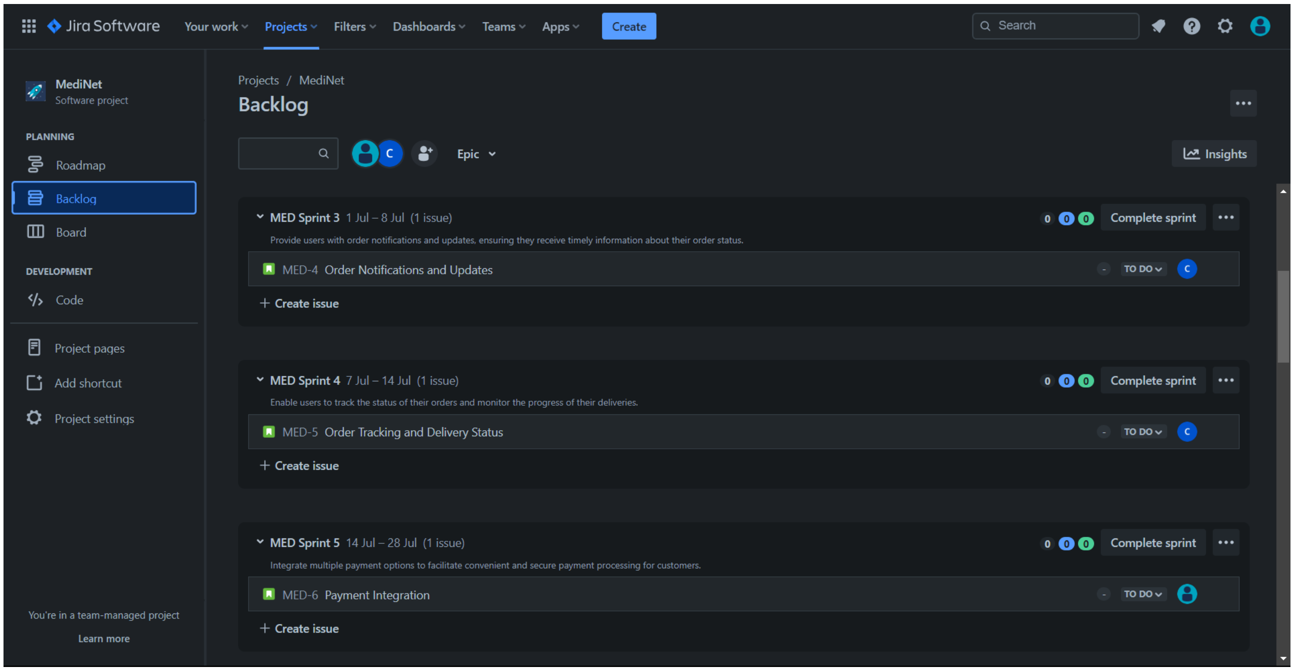
Implement the ordering and cart functionality, enabling users to add medicines to their cart and place orders seamlessly and verfying the prescription so that they get geniuen medicines

MED-3	Ordering and Cart Functionality	TO DO	C
MED-19	Presciption Verification	TO DO	

MED Sprint 3 1 Jul – 8 Jul (1 issue) 0 0 0

Provide users with order notifications and updates, ensuring they receive timely information about their order status.

MED-4	Order Notifications and Updates	TO DO	C



Projects / MediNet

Backlog

MED Sprint 3 1 Jul – 8 Jul (1 issue) 0 0 0 Complete sprint

Provide users with order notifications and updates, ensuring they receive timely information about their order status.

MED-4 Order Notifications and Updates - TO DO C

+ Create issue

MED Sprint 4 7 Jul – 14 Jul (1 issue) 0 0 0 Complete sprint

Enable users to track the status of their orders and monitor the progress of their deliveries.

MED-5 Order Tracking and Delivery Status - TO DO C

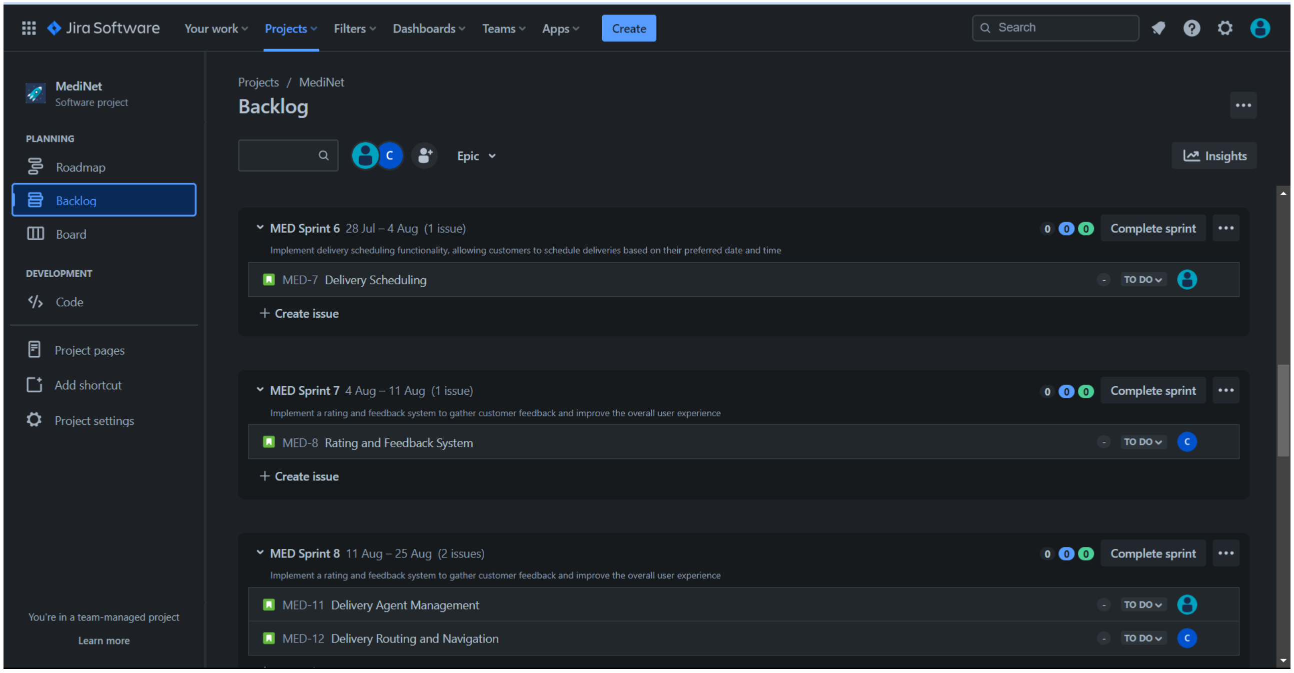
+ Create issue

MED Sprint 5 14 Jul – 28 Jul (1 issue) 0 0 0 Complete sprint

Integrate multiple payment options to facilitate convenient and secure payment processing for customers.

MED-6 Payment Integration - TO DO

+ Create issue



Projects / MediNet

Backlog

MED Sprint 6 28 Jul – 4 Aug (1 issue) 0 0 0 Complete sprint

Implement delivery scheduling functionality, allowing customers to schedule deliveries based on their preferred date and time

MED-7 Delivery Scheduling - TO DO

+ Create issue

MED Sprint 7 4 Aug – 11 Aug (1 issue) 0 0 0 Complete sprint

Implement a rating and feedback system to gather customer feedback and improve the overall user experience

MED-8 Rating and Feedback System - TO DO C

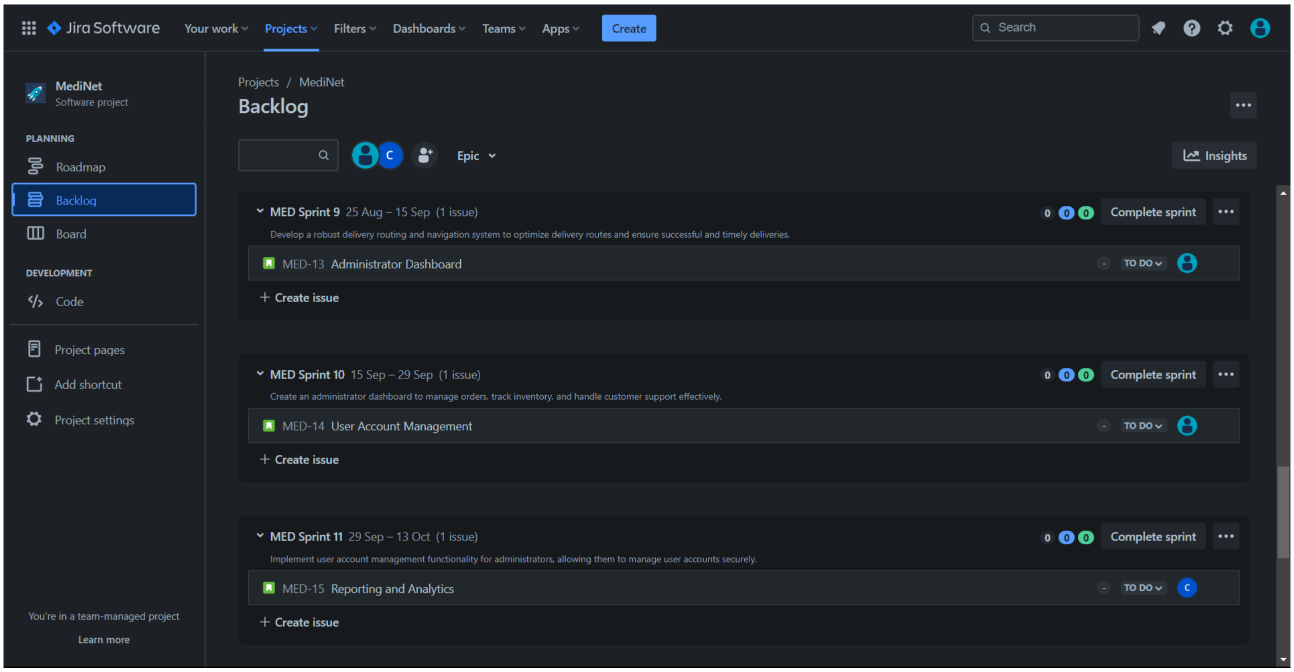
+ Create issue

MED Sprint 8 11 Aug – 25 Aug (2 issues) 0 0 0 Complete sprint

Implement a rating and feedback system to gather customer feedback and improve the overall user experience

MED-11 Delivery Agent Management - TO DO

MED-12 Delivery Routing and Navigation - TO DO C



Projects / MediNet

Backlog

MED Sprint 9 25 Aug – 15 Sep (1 issue) 0 0 0 Complete sprint

Develop a robust delivery routing and navigation system to optimize delivery routes and ensure successful and timely deliveries.

MED-13 Administrator Dashboard - TO DO

+ Create issue

MED Sprint 10 15 Sep – 29 Sep (1 issue) 0 0 0 Complete sprint

Create an administrator dashboard to manage orders, track inventory, and handle customer support effectively.

MED-14 User Account Management - TO DO

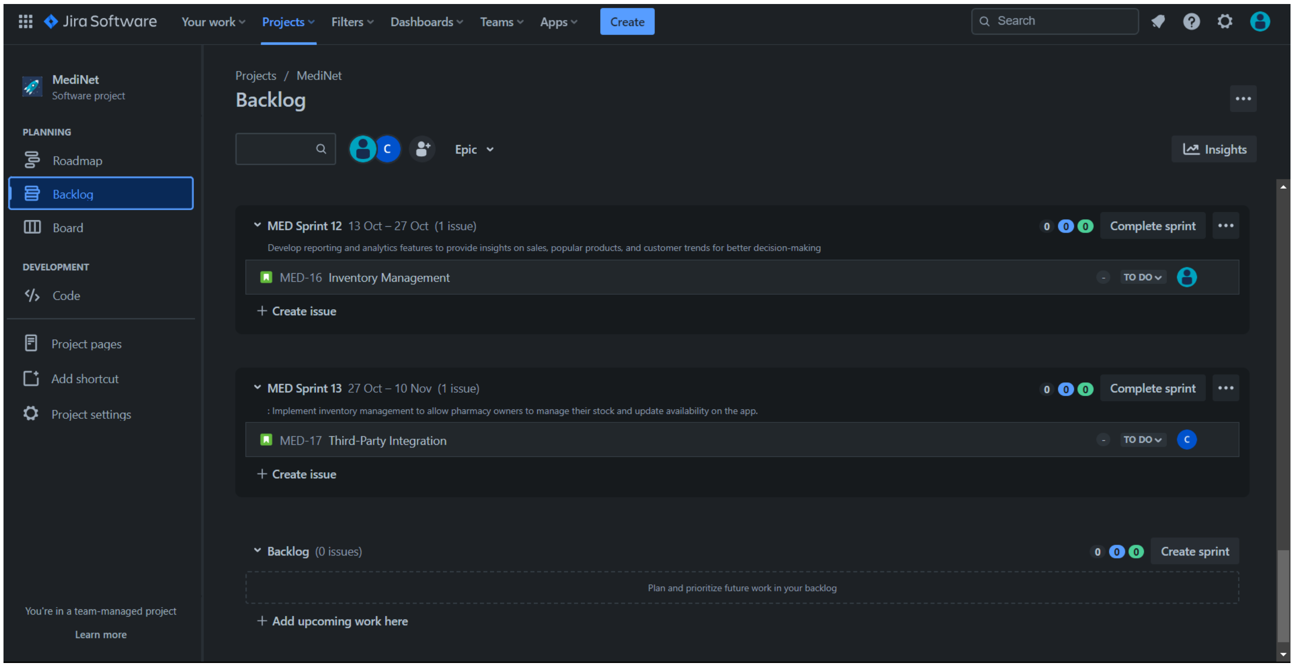
+ Create issue

MED Sprint 11 29 Sep – 13 Oct (1 issue) 0 0 0 Complete sprint

Implement user account management functionality for administrators, allowing them to manage user accounts securely.

MED-15 Reporting and Analytics - TO DO C

+ Create issue



Projects / MediNet

Backlog

MED Sprint 12 13 Oct – 27 Oct (1 issue) 0 0 0 Complete sprint

Develop reporting and analytics features to provide insights on sales, popular products, and customer trends for better decision-making

MED-16 Inventory Management - TO DO

+ Create issue

MED Sprint 13 27 Oct – 10 Nov (1 issue) 0 0 0 Complete sprint

: Implement inventory management to allow pharmacy owners to manage their stock and update availability on the app.

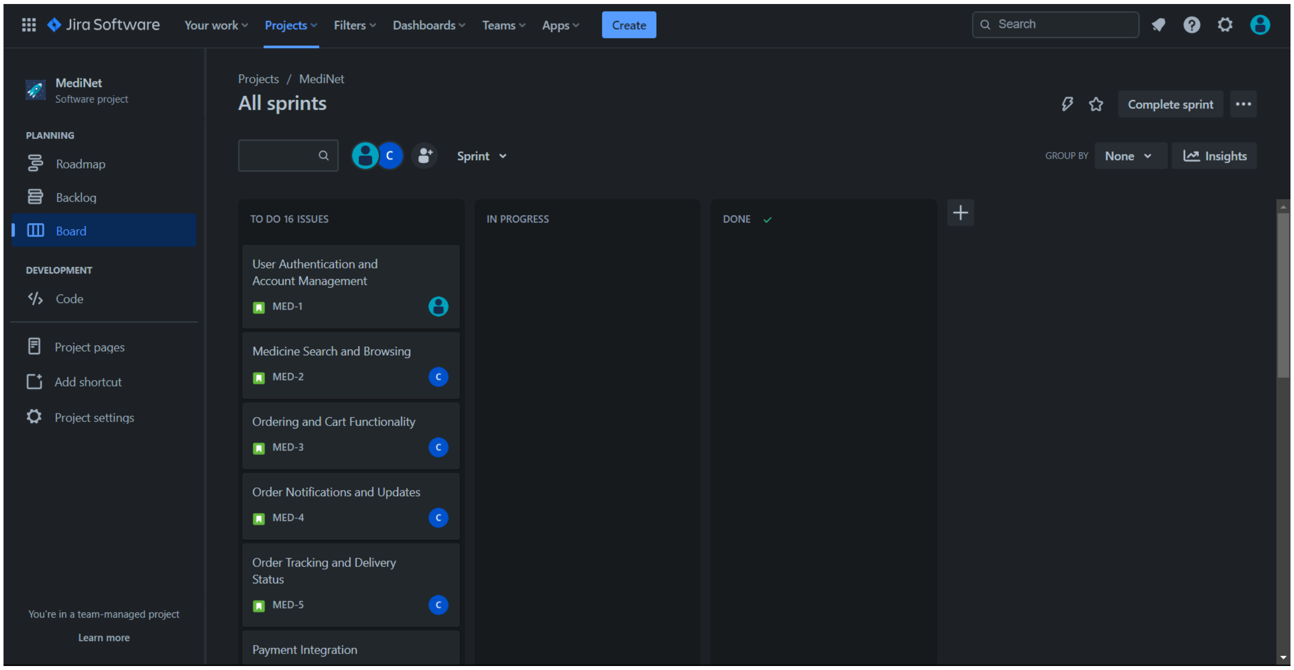
MED-17 Third-Party Integration - TO DO C

+ Create issue

Backlog (0 issues) 0 0 0 Create sprint

Plan and prioritize future work in your backlog

+ Add upcoming work here



User Authentication and Account Management
MED-1

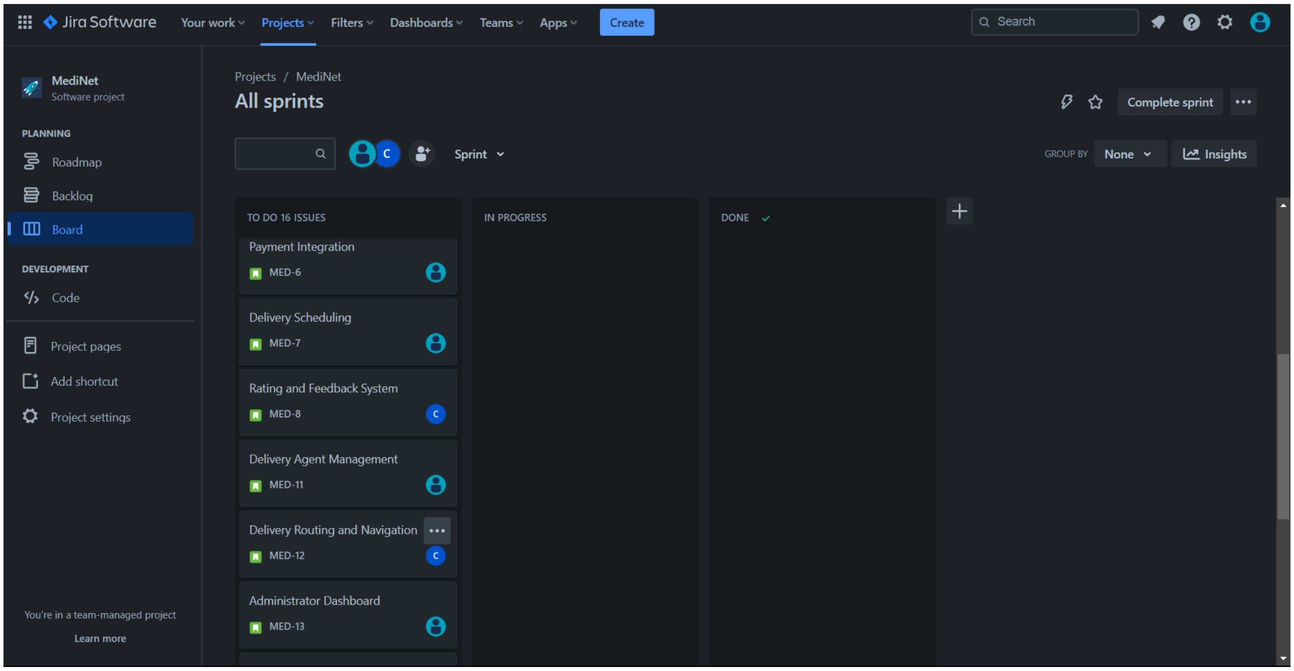
Medicine Search and Browsing
MED-2

Ordering and Cart Functionality
MED-3

Order Notifications and Updates
MED-4

Order Tracking and Delivery Status
MED-5

Payment Integration



Payment Integration
MED-6

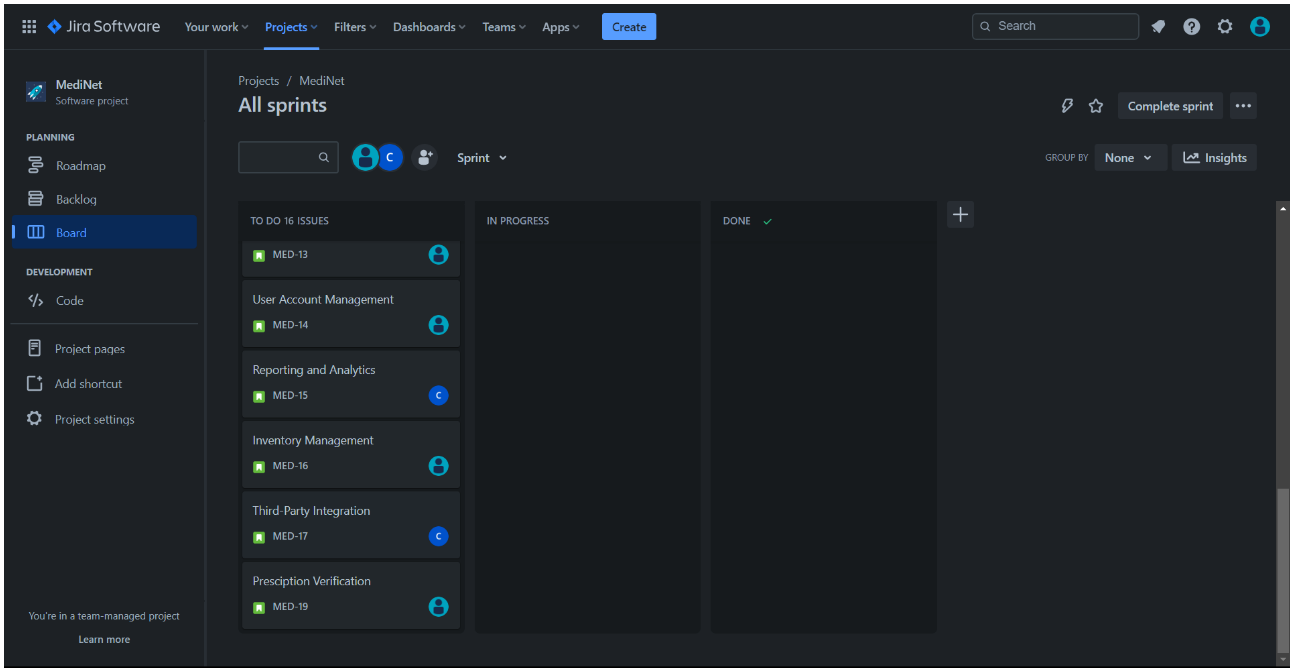
Delivery Scheduling
MED-7

Rating and Feedback System
MED-8

Delivery Agent Management
MED-11

Delivery Routing and Navigation
MED-12

Administrator Dashboard
MED-13



All sprints

MED-13

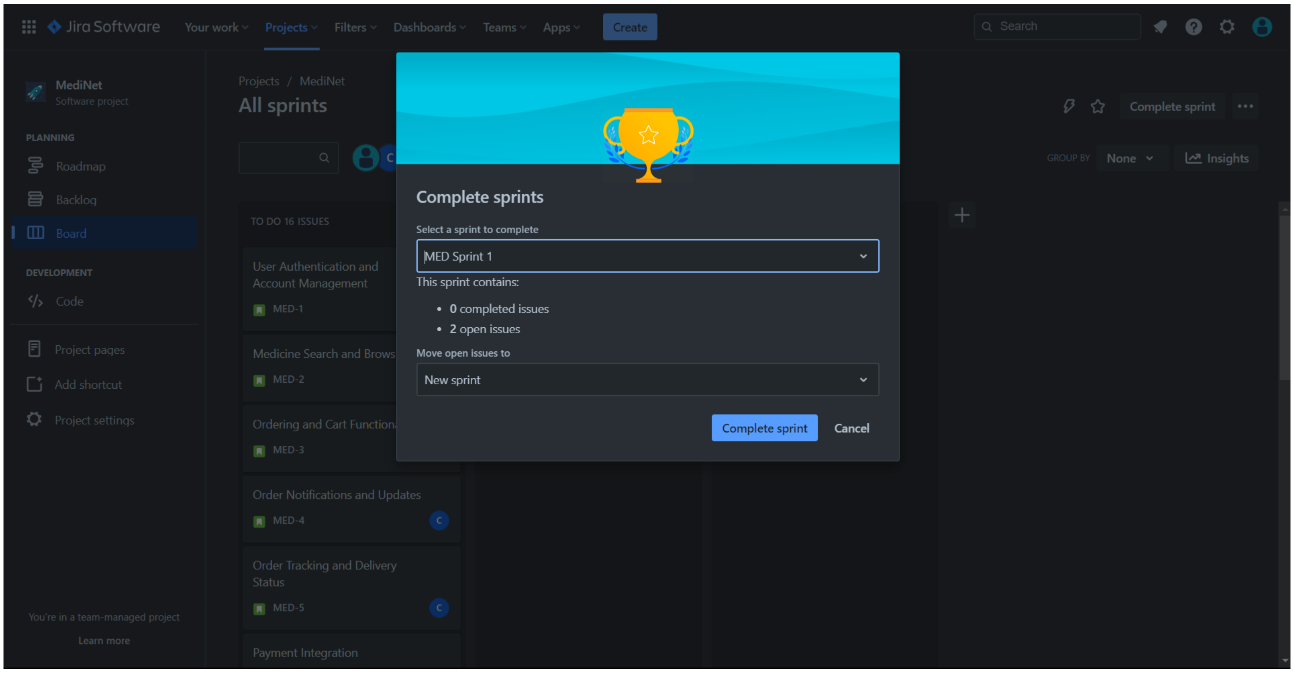
User Account Management
MED-14

Reporting and Analytics
MED-15

Inventory Management
MED-16

Third-Party Integration
MED-17

Presciption Verification
MED-19



All sprints

User Authentication and Account Management
MED-1

Medicine Search and Brows
MED-2

Ordering and Cart Function
MED-3

Order Notifications and Updates
MED-4

Order Tracking and Delivery Status
MED-5

Payment Integration

Complete sprints

Select a sprint to complete: MED Sprint 1

This sprint contains:

- 0 completed issues
- 2 open issues

Move open issues to: New sprint

Result:

A scrum sprint was created with user stories and it was completed when all the tasks were done.