**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**

****

**BÀI TẬP LỚN**

**TRÌNH BÀY CÁC PHƯƠNG PHÁP PHÁT TRIỂN PHẦN MỀM**

**QUẢN LÝ VÀ THEO DÕI LỖI WEBSITE GIỚI THIỆU TRÀ BẰNG CÔNG CỤ JIRA**

NGÀNH: CÔNG NGHỆ THÔNG TIN

CHUYÊN NGÀNH: KỸ THUẬT PHẦN MỀM

SINH VIÊN: **TRẦN THỊ MỸ TÂM**

MÃ LỚP: 12521161

HƯỚNG DẪN: **Th.S NGÔ THANH HUYỀN**

NHẬN XÉT

Nhận xét của giảng viên hướng dẫn:

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

**GIẢNG VIÊN HƯỚNG DẪN**

*(Ký và ghi rõ họ tên)*

# **LỜI CAM ĐOAN**

Em xin cam đoan bài tập lớp môn Kiểm thử và Đảm bào chất lượng phần mềm có tên “Trình bày các phương pháp phát triển phần mềm. Quản lý và theo dõi lỗi website giới thiệu trà bằng công cụ Jira.” là sản phẩm của bản thân. Những phần sử dụng tài liệu tham khảo trong bài tập lớn đã được nêu rõ trong phần tài liệu tham khảo. Các số liệu, kết quả trình bày trong bài tập lớn là hoàn toàn trung thực, nếu sai em xin chịu hoàn toàn trách nhiệm và chịu mọi kỷ luật của bộ môn và nhà trường đề ra.

|  |  |
| --- | --- |
|  | *Hưng Yên, ngày tháng năm 2023*  Sinh viên  Trần Thị Mỹ Tâm |

# **LỜI CẢM ƠN**

Để có thể hoàn thành đồ án này, đầu tiên em xin phép gửi lời cảm ơn tới bộ môn Công nghệ phần mềm, Khoa Công nghệ thông tin - Trường Đại học Sư phạm Kỹ thuật Hưng Yên đã tạo điều kiện thuận lợi cho em thực hiện đồ án môn học này.

Đặc biệt em xin chân thành cảm ơn thầy/cô Ngô Thanh Huyền đã rất tận tình hướng dẫn, chỉ bảo em trong suốt thời gian thực hiện bài tập lớn vừa qua.

Em cũng xin chân thành cảm ơn tất cả các Thầy, các Cô trong Trường đã tận tình giảng dạy, trang bị cho em những kiến thức cần thiết, quý báu để giúp em thực hiện được đồ án này.

Mặc dù em đã có cố gắng, nhưng với kiến thức còn hạn chế, trong quá trình thực hiện đề tài không tránh khỏi những thiếu sót. Em hi vọng sẽ nhận được những ý kiến nhận xét, góp ý của các Thầy giáo, Cô giáo về những kết quả triển khai trong đồ án.

Em xin trân trọng cảm ơn!

# **MỤC LỤC**

[NHẬN XÉT 2](#_Toc150468076)

[LỜI CAM ĐOAN 3](#_Toc150468077)

[LỜI CẢM ƠN 4](#_Toc150468078)

[MỤC LỤC 5](#_Toc150468079)

[DANH MỤC CÁC THUẬT NGỮ 7](#_Toc150468080)

[DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ 8](#_Toc150468081)

[CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI 9](#_Toc150468082)

[1.1 Lý do chọn đề tài 9](#_Toc150468083)

[1.2 Mục tiêu dự án 9](#_Toc150468084)

[1.2.1 Mục tiêu tổng quát 9](#_Toc150468085)

[1.2.2 Mục tiêu cụ thể 9](#_Toc150468086)

[1.3 Giới hạn và phạm vi đề tài 10](#_Toc150468087)

[1.3.1 Đối tượng nghiên cứu 10](#_Toc150468088)

[1.3.2 Phạm vi nghiên cứu 10](#_Toc150468089)

[1.4 Nội dung thực hiện 10](#_Toc150468090)

[1.5 Phương pháp tiếp cận 10](#_Toc150468091)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 12](#_Toc150468092)

[2.1 Các phương pháp phát triển phần mềm 12](#_Toc150468093)

[2.1.1 Mô hình thác nước- Waterfall 12](#_Toc150468094)

[2.1.2 Mô hình Kanban 13](#_Toc150468095)

[2.1.3 Mô hình Spiral 15](#_Toc150468096)

[2.1.4 Mô hình Agile 16](#_Toc150468097)

[2.1.5 Mô hình V-Model 17](#_Toc150468098)

[2.1.6 Mô hình Crum 19](#_Toc150468099)

[2.1.7 Mô hình Extreme Programming (XP) 20](#_Toc150468100)

[2.2 Quản lý và theo dõi lỗi phần mềm bằng công cụ Jira 21](#_Toc150468101)

[2.2.1 Jira là gì? 21](#_Toc150468102)

[2.2.2 Các tính năng chính của Jira bao gồm: 22](#_Toc150468103)

[2.2.3 Quy trình quản lý và theo dõi lỗi bằng Jira: 22](#_Toc150468104)

[2.2.4 Ưu nhược điểm của Jira 23](#_Toc150468105)

[2.2.5 Các thành phần của Jira 23](#_Toc150468106)

[CHƯƠNG 3: NỘI DUNG THỰC HIỆN 25](#_Toc150468107)

[3.1 Cài đặt 25](#_Toc150468108)

[3.2 Cách sử dụng Jira 25](#_Toc150468109)

[3.2.1 Đăng nhập vào Jira (Đối với Web) 25](#_Toc150468110)

[3.2.2 Một số thao tác cơ bản 26](#_Toc150468111)

[3.2.3 Tạo mới 1 issue 29](#_Toc150468112)

[3.2.4. Thao tác với 1 issue 31](#_Toc150468113)

[3.2.5. Tìm kiếm, lọc issue 33](#_Toc150468114)

[3.3 Áp dụng vào dự án cụ thể (Đồ án 3) 34](#_Toc150468115)

[3.3.1 Tạo tài khoản miễn phí trên Atlassian 34](#_Toc150468116)

[CHƯƠNG 4: KẾT LUẬN 35](#_Toc150468117)

[4.1 Kết quả đạt được 35](#_Toc150468118)

[4.2 Hướng phát triển đề tài 35](#_Toc150468119)

[TÀI LIỆU THAM KHẢO 36](#_Toc150468120)

DANH MỤC CÁC THUẬT NGỮ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Từ viết tắt | Cụm từ tiếng anh | Diễn giải |
| 1 | SRS | Software Requirement Specification | Đặc tả yêu cầu phần mềm |
| 2 | PM | Project Manager |  |
| 3 | QA | Quality Assurance |  |
| 4 | Dev | Developer |  |

# **DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ**

[Hình 2. 1 Mô hình thác nước-waterfall 10](#_Toc150467461)

[Hình 2. 2 Mô hình Kanban 11](#_Toc150467462)

[Hình 2. 3 Mô hình xoắn ốc-Spiral 13](#_Toc150467463)

[Hình 2. 4 Mô hình agile 14](#_Toc150467464)

[Hình 2. 5 Mô hình V-Model 16](#_Toc150467465)

[Hình 2. 6 Mô hình crum 17](#_Toc150467466)

[Hình 2. 7 Mô hình Extreme Programming (XP) 18](#_Toc150467467)

[Hình 3. 1 Đăng nhập 23](#_Toc150467468)

[Hình 3. 2 Sau khi đăng nhập 24](#_Toc150467469)

[Hình 3. 3 Thu hẹp menu 25](#_Toc150467470)

[Hình 3. 4 Hiển thị danh sách các dự án 25](#_Toc150467471)

[Hình 3. 5 Mở Submenu 26](#_Toc150467472)

[Hình 3. 6 Tạo 1 issue mới 27](#_Toc150467473)

[Hình 3. 7 Thao tác với 1 issue 29](#_Toc150467474)

[Hình 3. 8 Tìm kiếm, lọc issue 31](#_Toc150467475)

# **CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI**

* 1. **Lý do chọn đề tài**

Với sự phát triển như vũ bão của công nghệ thông tin nói chung và công nghệ phần mềm nói riêng, việc phát triển phần mềm ngày càng được hỗ trợ bởi nhiều công cụ tiên tiến, giúp cho việc xây dựng phần mềm đỡ mệt nhọc và hiệu quả hơn. Tuy nhiên, vì độ phức tạp của phần mềm và những giới hạn về thời gian và chi phí nên không đảm bảo được rằng các sản phẩm phần mềm đang được ứng dụng không có lỗi. Vì vậy, áp dụng kiểm thử và đảm bảo chất lượng phần mềm bằng một công cụ quản lý lỗi Bugmantis thực sự rất cần thiết, để đảm bảo rằng phần mềm thỏa mãn các yêu cầu thiết kế và các yêu cầu đó đáp ứng các nhu cầu của người dùng.

Quản lí dự án, kiểm soát và theo dõi lỗi luôn là những vấn đề vô cùng quan trọng và rất được quan tâm trong các dự án. Hiện nay, để hỗ trợ cho việc kiểm thứ phần mềm đã có rất nhiều các công cụ ra đời nhằm quản lí hiệu quả bug giúp ích không nhỏ cho công việc của những chuyên viên kiểm thử phần mềm. Ví dụ có thể kể đến như Bugmantis, Bugherd, ...Trong đó phải kể đến Jira - một công cụ quản lí bug rất phổ biến hiện nay. Trong phần này , mình sẽ trình bày một số kiến thức tổng quan và hướng dẫn sử dụng cơ bản về phần mềm này.

* 1. **Mục tiêu dự án**
     1. ***Mục tiêu tổng quát***

Quản lý và theo dõi lỗi phần mềm bằng công cụ Jira

* + 1. ***Mục tiêu cụ thể***

Sử dụng hệ thống quản lý dự án“Xây dựng website giới thiệu trà”, theo dõi trên Jira.

Quản lý các chức năng:

* Đăng nhập
* Đăng ký
* Quản lý sản phẩm
* Quản lý giỏ hàng
* Quản lý đặt hàng

…

* 1. **Giới hạn và phạm vi đề tài**
     1. ***Đối tượng nghiên cứu***

Đối tượng nghiên cứu: Website giới thiệu trà.

* + 1. ***Phạm vi nghiên cứu***

Áp dụng cho tất cả các dự án trong nội bộ công ty, nhà trường...

Áp dụng cho các dự án khách hàng yêu cầu sử dụng hệ thống theo dõi và kiểm soát công việc, lỗi phần mềm.

Ý nghĩa khoa học và thực tiễn của đề tài :

- Quản lý các loại vấn đề: nhiệm vụ , lỗi, yêu cầu hỗ trợ, .. của dự án một cách cơ bản.

- Hỗ trợ làm việc nhóm, nâng cao hiệu quả, trình độ teamwork.

* 1. **Nội dung thực hiện**
* Cần cài đặt công cụ quản lý lỗi
* Biết cách sử dụng công cụ
* Thực hiện kiểm thử website
* Quản lý dự án, issue

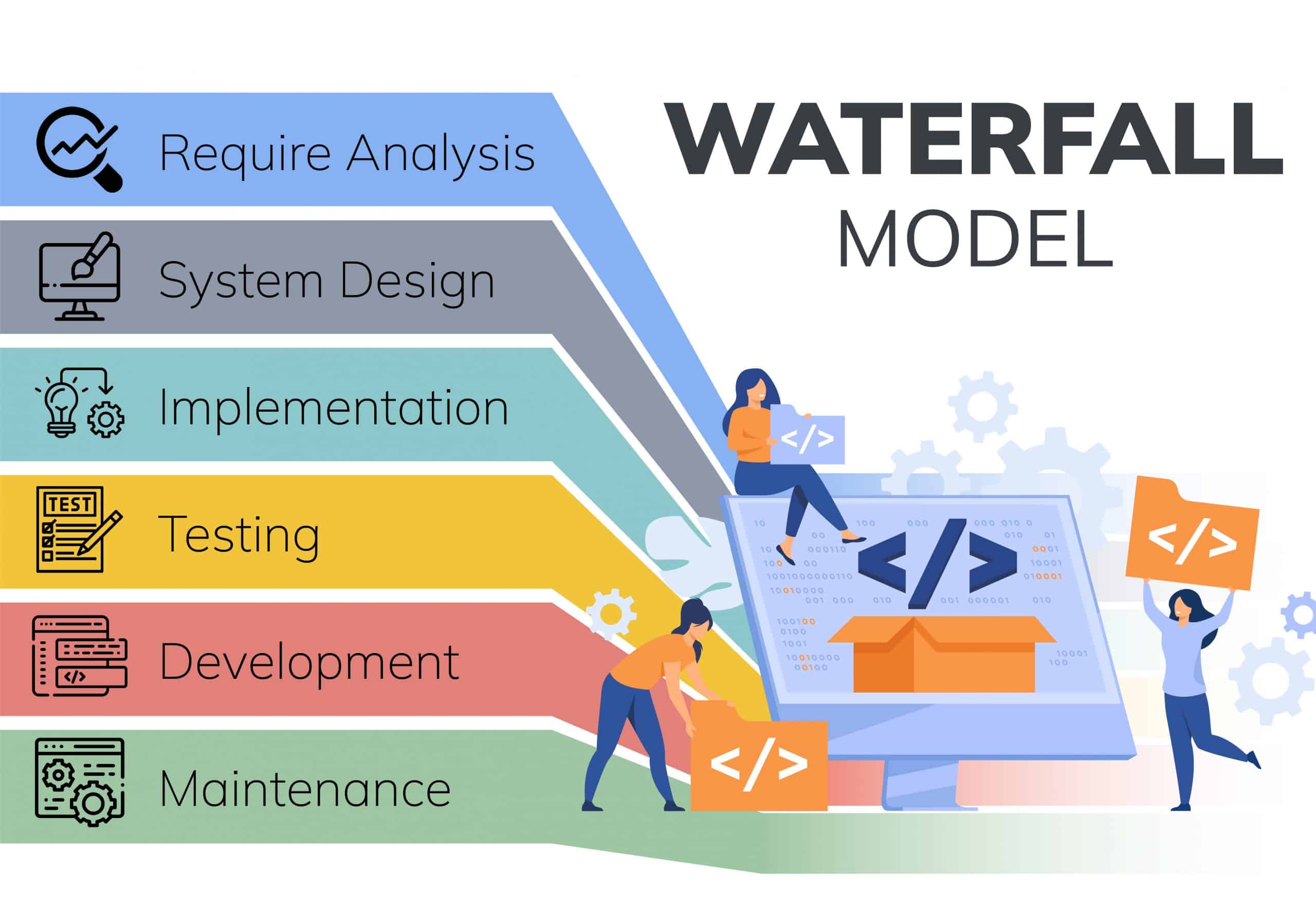
Sau khi bàn bạc và thống nhất trong nhóm thì công việc của em đảm nhiệm là quản lý website bán xe máy MY HONDA:

* Đăng nhập
* Đăng ký
* Quản lý sản phẩm
  1. Phương pháp tiếp cận
* Cách tiếp cận : Nghiên cứu và sử dụng Phần mềm quản lý lỗi Jira
* Sử dụng các phương pháp nghiên cứu:
  + Phương pháp đọc tài liệu;
  + Phương pháp phân tích mẫu;
  + Phương pháp thực nghiệm.

# **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

## **2.1 Các phương pháp phát triển phần mềm**

### **2.1.1 Mô hình thác nước- Waterfall**



Hình 2. 1 Mô hình thác nước-waterfall

Phương pháp watefall là một phương pháp quản lý truyền thống không còn phổ biến trong việc phát triển phần mềm nữa. Trong nhiều năm trước, mô hình waterfall là một trong những phương pháp quản lý hàng đầu, nhưng cách tiếp cận cứng nhắc của nó không thể đáp ứng nhu cầu năng động của phát triển phần mềm ở thời điểm hiện tại.

Người ta thường thấy phương pháp waterfall được sử dụng để quản lý dự án hơn là phát triển sản phẩm. Khi bắt đầu một dự án, các nhà quản lý dự án thu thập tất cả các thông tin cần thiết và sử dụng nó để lập một kế hoạch hành động cụ thể với tất cả các thông tin liên quan. Thông thường, kế hoạch này là một quy trình tuyến tính, từng bước với các nhiệm vụ được sắp xếp liên tục, đó chính là lý do mà phương pháp này mang tên “waterfall”

Phương pháp waterfall được định hướng theo kế hoạch một cách cứng nhắc, để lại rất ít chỗ để điều chỉnh. Nó ít nhiều ngược lại với agile, ưu tiên bám sát kế hoạch hơn là thích nghi với hoàn cảnh mới.

**- Ưu điểm:**

- Dễ hiểu và triển khai.

- Phù hợp cho các dự án có yêu cầu ổn định và rõ ràng.

- Phù hợp cho các dự án nhỏ hoặc đơn giản.

- Thích hợp cho các dự án có khả năng mở rộng thấp.

**- Nhược điểm:**

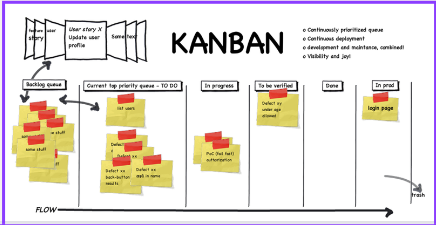
- Khó thích nghi với thay đổi trong yêu cầu.

- Không phù hợp cho các dự án lớn và phức tạp.

- Khách hàng thường chỉ có cơ hội xem sản phẩm ở cuối dự án.

- Có thể gây ra việc lãng phí thời gian và nguồn lực nếu cần thay đổi sau khi đã bắt đầu.

### **2.1.2 Mô hình Kanban**



Hình 2. 2 Mô hình Kanban

Kanban cũng là một phương pháp Agile, tập trung vào quản lý công việc dưới dạng một luồng liên tục. Các công việc được đưa vào bảng Kanban và di chuyển qua các cột từ trạng thái "đang làm" đến "hoàn thành."

Phương pháp Kanban là một phương pháp quản lý và phát triển dự án tập trung vào quy trình làm việc và kiểm soát công việc trong một luồng liên tục. Dưới đây là ưu điểm và nhược điểm của phương pháp Kanban:

**-Ưu điểm:**

-Tối ưu hóa quy trinh

-Tích hợp dễ dàng

-Tập trung vào công việc quan trọng

-Phản hồi nhanh chông

-Dễ theo dõi tiến độ

**-Nhược điểm:**

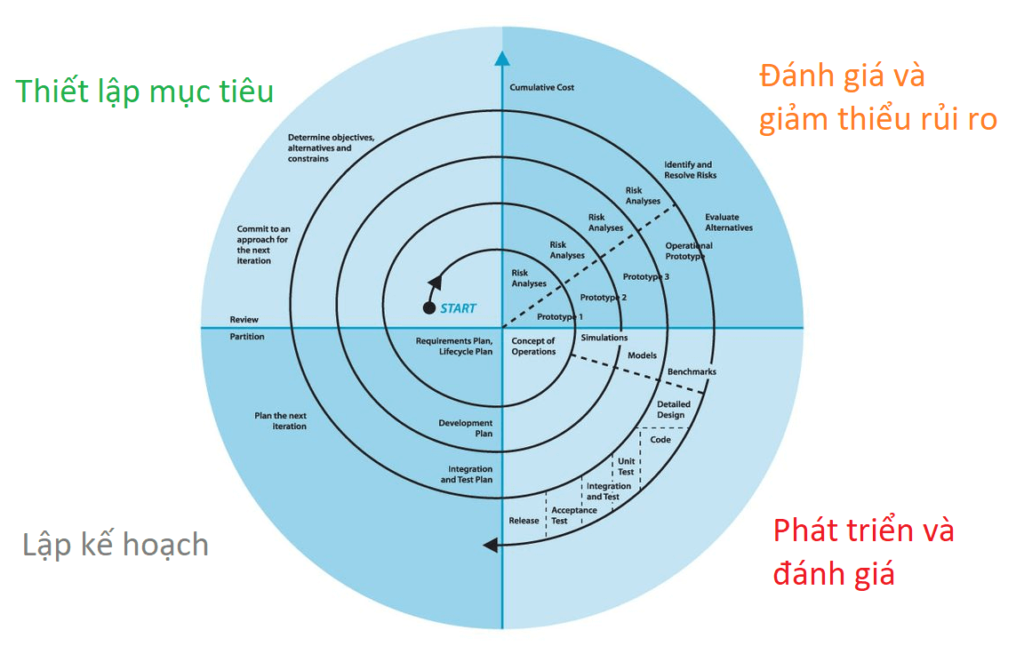
-Khó kiểm soát cho các dự án lớn

-Khó đánh giá thời gian hoàn thành

-Yêu cầu tự quản lý

-Khó phát triển trong lớn.

### **2.1.3 Mô hình Spiral**



Hình 2. 3 Mô hình xoắn ốc-Spiral

Mô hình này kết hợp các khía cạnh của phát triển phần mềm tuần tự và linh hoạt. Nó tập trung vào việc đánh giá rủi ro và kiểm soát chúng trong suốt quá trình phát triển.

Mô hình Spiral (Spiral Model) là một phương pháp phát triển phần mềm kết hợp các khía cạnh của phát triển tuần tự và linh hoạt. Nó được sử dụng để quản lý các dự án phức tạp và có khả năng đánh giá và kiểm soát rủi ro. Dưới đây là ưu điểm và nhược điểm của Mô hình Spiral:

**-Ưu điểm:**

-Quản lý rủi ro tốt

-Phù hợp cho các dự án phức tạp

-Tối ưu hóa sự linh hoạt

-Kiểm soát chất lượng

**-Nhược điểm**

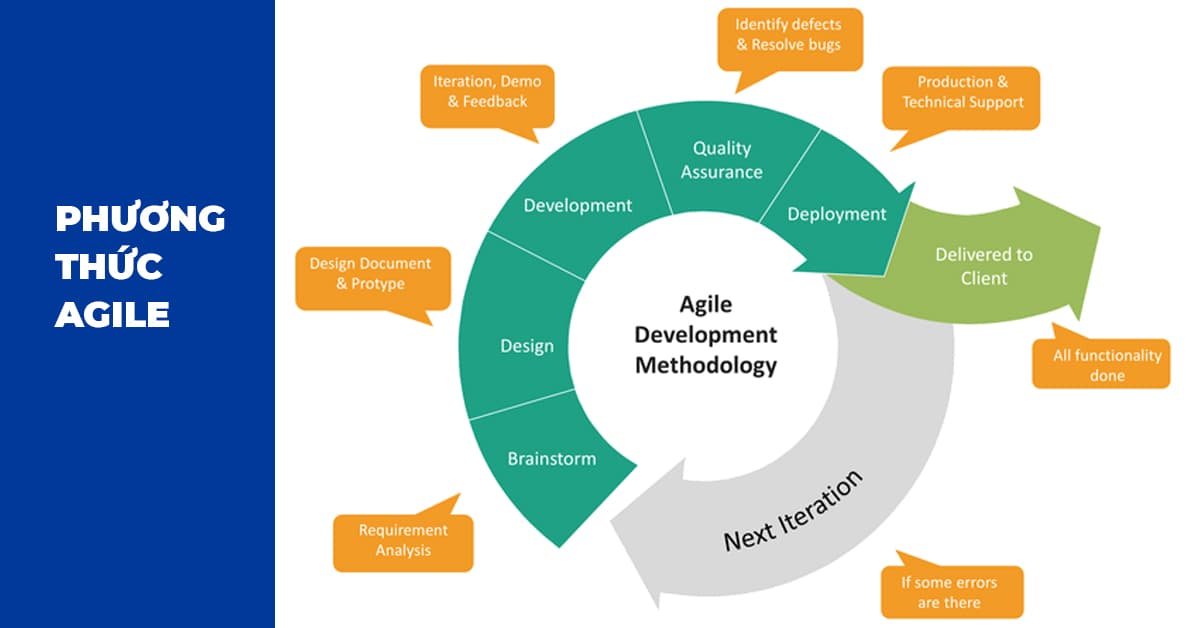
-Phức tạp trong quản lý

-Tốn thời gian và nguồn lực

-Khó hiểu và triển khai

-Không phù hợp cho các dự án nhỏ

### **2.1.4 Mô hình Agile**



Hình 2. 4 Mô hình agile

Agile là một khái niệm thông dụng nhất dùng dể diễn tả việc phát triển phần mềm. Nó có thể được xem như là một từ khóa trong mọi lĩnh vực mà cần phải làm thật nhanh, điều này cũng có nghĩa agile nhấn mạnh vào việc tránh lãng phí thời gian và tối ưu hóa công suất làm việc.

Phần lớn các phương pháp phát triển phần mềm khác đều dựa trên agile với việc nhấn mạnh vào sự lặp lại, cộng tác người với người, đề cao tính hiệu quả trong công việc, những điều này trái ngược với các phương pháp phát triển cũ. Một ví dụ khá hay mà tôi dùng để hính tượng hóa sự so sánh đó là agile và phương pháp cũ giống như là nhạc jazz và nhạc cổ điển.

Các phương pháp quản lý theo kiểu truyền thống, chẳng hạn như phương pháp waterfall mà tôi sẽ đề cập bên dưới, giống như nhạc cổ điển, do một người chỉ huy có kế hoạch cụ thể về cách phát nhạc. Mặt khác, agile giống như nhạc jazz, kết hợp với nhau thông qua sự hợp tác, thử nghiệm và lặp đi lặp lại giữa các thành viên trong ban nhạc. Agile thích ứng và phát triển với những ý tưởng, tình huống và hướng đi mới.

**- Ưu điểm:**

- Linh hoạt và thích ứng với thay đổi yêu cầu.

- Tương tác thường xuyên với khách hàng, giúp đảm bảo sự hài lòng của họ.

- Có thể phát triển sản phẩm từng phần một và đưa ra thị trường nhanh chóng.

- Tạo cơ hội để cải thiện liên tục.

**- Nhược điểm:**

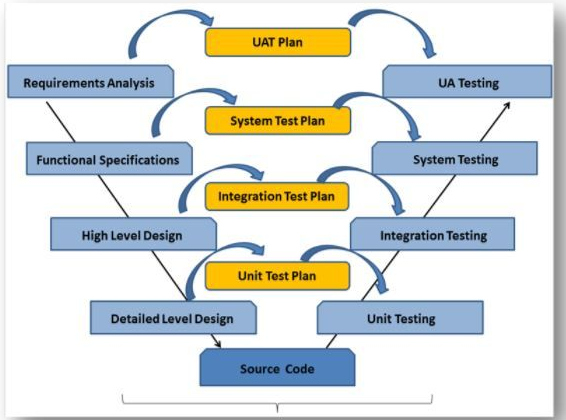
- Khó quản lý trong các dự án lớn và phức tạp.

- Yêu cầu sự cam kết và tham gia tích cực từ phía khách hàng.

- Cần có sự quản lý kỹ thuật tốt để đảm bảo chất lượng.

- Yêu cầu sự linh hoạt và tự quản lý từ nhóm phát triển.

**2.1.5 Mô hình V-Model**

****

Hình 2. 5 Mô hình V-Model

Mô hình V hiện nay là một trong những quy trình phát triển phần mềm được sử dụng rộng rãi nhất. Trong mô hình V việc thực hiện kiểm tra được diễn ra ngay từ giai đoạn lấy yêu cầu. V mô hình cũng được gọi là mô hình xác minh (verification) và mô hình xác nhận (validation). Mô hình V-Model kết hợp các giai đoạn phát triển với các giai đoạn kiểm thử tương ứng. Nó tập trung vào việc đảm bảo chất lượng phần mềm thông qua việc kiểm thử từng giai đoạn.

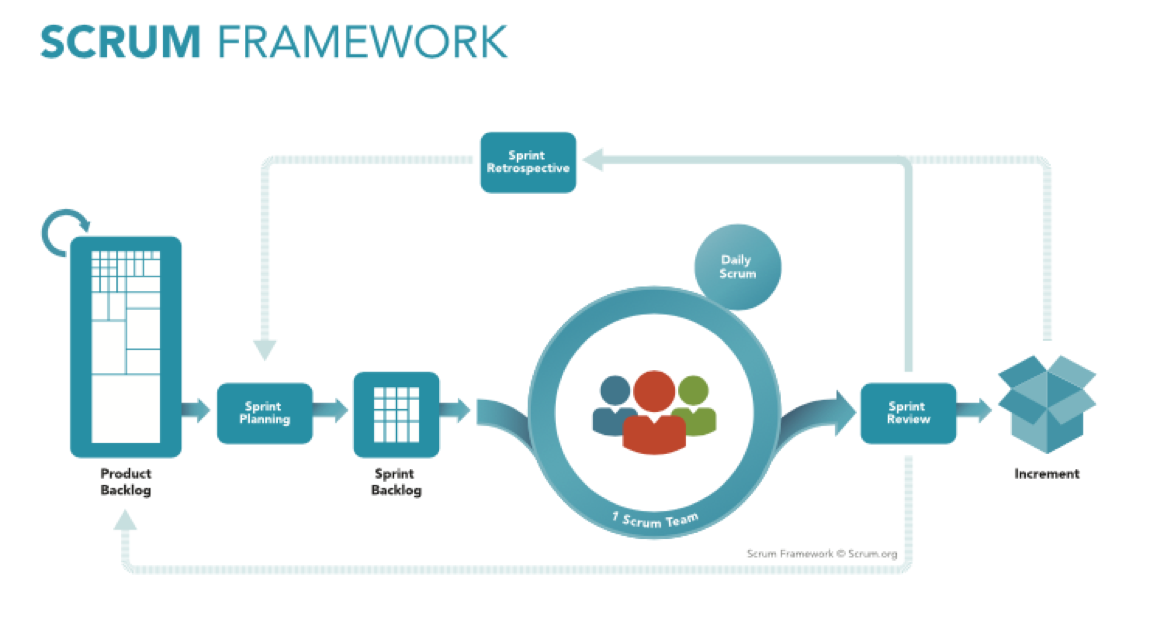
1. Ưu điểm:

* Quá trình phát triển và quy trình quản lý có tính tổ chức và hệ thống
* Hoạt động tốt cho các dự án có quy mô vừa và nhỏ.
* Kiểm tra bắt đầu từ khi bắt đầu phát triển vì vậy sự mơ hồ được xác định ngay từ đầu.
* Dễ dàng quản lý vì mỗi giai đoạn có các mục tiêu và mục tiêu được xác định rõ ràng.

2. Nhược điểm:

* Không thích hợp cho các dự án lớn và phức tạp
* Không phù hợp nếu các yêu cầu thường xuyên thay đổi.
* Không có phần mềm làm việc được sản xuất ở giai đoạn trung gian.
* Không có điều khoản cho việc phân tích rủi ro nên có sự không chắc chắn và có tính rủi ro.

### **2.1.6 Mô hình Crum**



Hình 2. 6 Mô hình crum

Scrum là một phương pháp Agile phát triển phần mềm tập trung vào việc quản lý và kiểm soát dự án thông qua các sprints (chu kỳ làm việc ngắn hạn) và các cuộc họp hàng ngày để theo dõi tiến độ.

**- Ưu điểm:**

- Tạo cơ hội cho việc kiểm soát thường xuyên và điều chỉnh.

- Phân chia dự án thành các sprint ngắn hạn, giúp quản lý thời gian tốt hơn.

- Tạo sự tham gia tích cực của các thành viên trong nhóm.

- Tập trung vào ưu tiên hóa công việc.

**- Nhược điểm:**

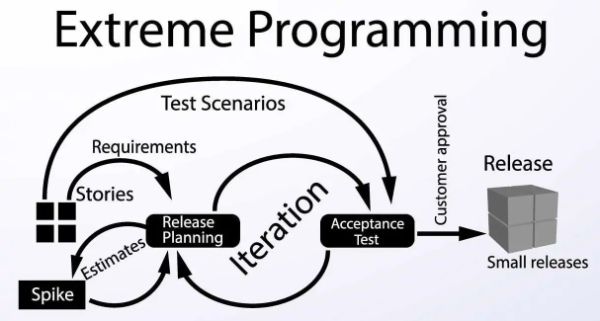
- Cần sự đầu tư và sự cam kết từ toàn bộ nhóm.

- Không phù hợp cho các dự án có yêu cầu không ổn định hoặc thay đổi thường xuyên.

- Yêu cầu sự quản lý kỹ thuật tốt.

- Không phù hợp cho các dự án lớn và phức tạp.

### **2.1.7 Mô hình Extreme Programming (XP)**



Hình 2. 7 Mô hình Extreme Programming (XP)

XP là một phương pháp phát triển phần mềm Agile với việc tập trung vào chất lượng mã nguồn và tương tác thường xuyên với khách hàng. Nó thúc đẩy các thực hành như kiểm thử liên tục, lập trình theo cặp, và lập kế hoạch ngắn hạn.

Extreme Programming (XP) là một phương pháp phát triển phần mềm Agile tập trung vào chất lượng mã nguồn và tương tác thường xuyên với khách hàng. Dưới đây là ưu điểm và nhược điểm của phương pháp Extreme Programming (XP):

**-Ưu điểm:**

-Chất lượng mã nguồn cao

-Tương tác thường xuyên với khách hàng

-Phản hồi nhanh chông

-Tập trung vào giá trị

-Cộng đồng phát triển tích cực

**-Nhược điểm:**

-Yêu cầu sự cam kết cao

-Không phù hợp cho mọi dự án

-Khó khảo sát tiến độ

-Đòi hỏi sự hiểu biết về XP

## **Quản lý và theo dõi lỗi phần mềm bằng công cụ Jira**

### **2.2.1 Jira là gì?**

Jira là một phần mềm quản lý và theo dõi lỗi, vấn đề và công việc trong dự án phần mềm. Nó được phát triển bởi công ty Atlassian và được sử dụng rộng rãi trong ngành công nghiệp phần mềm.

Quản lý dự án, kiểm soát và theo dõi lỗi, kiểm soát và theo dõi Vấn đề phát sinh trong một tổ chức. Theo dõi và quản lý vấn đề và lỗi phát sinh trong quá trình của một dự án là một công việc tối quan trọng, nhưng rất ít các dự án được quản lý một cách hiệu quả.

JIRA là một ứng dụng theo dõi và quản lý lỗi, vấn đề và dự án, được phát triển để làm quy trình này trở nên dễ dàng hơn cho mọi tổ chức. JIRA đã được thiết kế với trọng tâm vào kết quả công việc, có thể sử dụng ngay và linh hoạt khi sử dụng.

Jira dựa trên 3 khái niệm:

1. Issue: Mọi task, lỗi, yêu cầu nâng cao; về cơ bản mọi thứ được tạo ra và theo dõi qua JIRA được coi là một vấn đề.
2. Dự án: tập hợp các issues
3. Luồng công việc: Luồng công việc đơn giản là một loạt các bước mà một vấn đề trải qua bắt đầu từ khi tạo thành đến khi hoàn thành.

**2.2.2 Các tính năng chính của Jira bao gồm:**

- Quản lý và theo dõi lỗi, vấn đề, công việc và nâng cấp.

- Tạo và quản lý các quy trình làm việc phù hợp với yêu cầu của từng dự án.

- Tìm kiếm nhanh chóng với JIRA Query Language.

- Tạo báo cáo và biểu đồ thống kê để theo dõi tiến độ và hiệu suất của dự án.

- Tích hợp dễ dàng với các hệ thống khác như Email, Excel và RSS.

* + 1. **Quy trình quản lý và theo dõi lỗi bằng Jira:**

1. Tạo và quản lý các dự án: Bắt đầu bằng việc tạo một dự án trong Jira để quản lý các lỗi và công việc liên quan.

2. Tạo và gán lỗi: Khi phát hiện lỗi, bạn có thể tạo một lỗi mới trong Jira và gán cho người phụ trách để giải quyết.

3. Theo dõi và cập nhật lỗi: Theo dõi tiến trình giải quyết lỗi và cập nhật trạng thái của lỗi trong Jira.

4. Giao tiếp và phản hồi: Sử dụng Jira để giao tiếp và phản hồi với các thành viên khác trong dự án về các vấn đề và lỗi.

5. Kiểm tra và xác nhận: Sau khi lỗi được giải quyết, kiểm tra và xác nhận lại để đảm bảo rằng lỗi đã được khắc phục đúng cách.

6. Báo cáo và thống kê: Tạo báo cáo và biểu đồ thống kê từ dữ liệu trong Jira để theo dõi tiến độ và hiệu suất của dự án.

* + 1. Ưu nhược điểm của Jira
* Ưu điểm.
* Jira có chức năng phân quyền cực kỳ chi tiết, không chỉ phân quyền trong dự án chung, mà còn phân quyền đối với từng nhiệm vụ, giúp team công nghệ bảo vệ thông tin độc quyền của mình.
* Dễ dàng tích hợp với các hệ thống khác (như Email, Excel, RSS,...)
* Hệ thống module và bộ công cụ phát triển bổ trợ cho phép tùy biến, mở rộng và tích hợp Jira vào trong hệ thống hiện tại
* Jira được phát triển sử dụng chuẩn HTML và được thử nghiệm với tất cả các trình duyệt phổ biến hiện nay.
* Có thể chạy trên hầu hết các nền tảng phần cứng, hệ điều hành và cơ sở dữ liệu
* Mỗi màn hình trong Jira có một phiên bản có thể in đảm bảo việc luân chuyển bản cứng một cách dễ dàng
* Có thể tích hợp trực tiếp với code trên môi trường phát triển, là một công cụ hoàn toàn phù hợp với các developer
* Nhược điểm.
* Chi phí cao, sau 7 ngày dùng thử thì doanh nghiệp càng có quy mô lớn thì càng tốn nhiều chi phí: $10 mỗi tháng dành cho tối đa 10 tài khoản; từ 11-100 tài khoản là $7/tài khoản/tháng
* Tốn nhiều thời gian và công sức để setup nên chỉ phát huy tối ưu hiệu quả với dự án lớn, không phù hợp với dự án vừa và nhỏ (dưới 3 tháng)
* Ngôn ngữ tiếng Anh với nhiều thuật ngữ khó sử dụng
* Quy trình làm việc phức tạp đòi hỏi phải tìm hiểu kỹ lưỡng
  + 1. **Các thành phần của Jira**
* Roles: Xác lập các role của dự án, Mục này xác nhận ai tham gia vào dự án, những người add vào role thì mới có thể tạo Resource Allocation và project team sau này. Nhiều người có thể vào 1 role.
* Issue: là các tasks, các bugs, các features hay bất kỳ các type khác của project work
* Project: Chức năng này dùng để phân quyền approve worklog cho thành viên của dự án. Ai là team lead của group nào thì sẽ được approve worklog cho member của group đó. Project management được quyền approve cho toàn bộ thành viên dự án. C
* Component là sản phẩm của dự án. Ở đây sẽ nhập tất cả sản phẩm của dự án lấy từ file kế hoạch doanh số. Nếu dự án làm theo Scrum thì sẽ là Product của Sprint tương ứng.
* Workflow: Là một quản trị JIRA, bạn có thể cấu hình gây nên quy trình làm việc, điều kiện, xác nhận, và sau chức năng. Trang này sẽ cung cấp một cái nhìn tổng quan và các bước cơ bản cho từng phần của công việc của bạn
* Priority: Là mức độ ưu tiên của một defect. Có 4 mức , chọn theo datalist
* Status: Đại diện cho các vị trí của vấn đề trong workflow
* Resolution: Một issue có thể được resolved theo nhiều cách, một trong số đó là ‘Fixed’. Một resolution thường gán khi status bị thay đổi. Các resolution mặc định được liệt kê bên dưới, nhưng người quản trị JIRA có thể tùy biến theo nhu cầu của tổ chức.

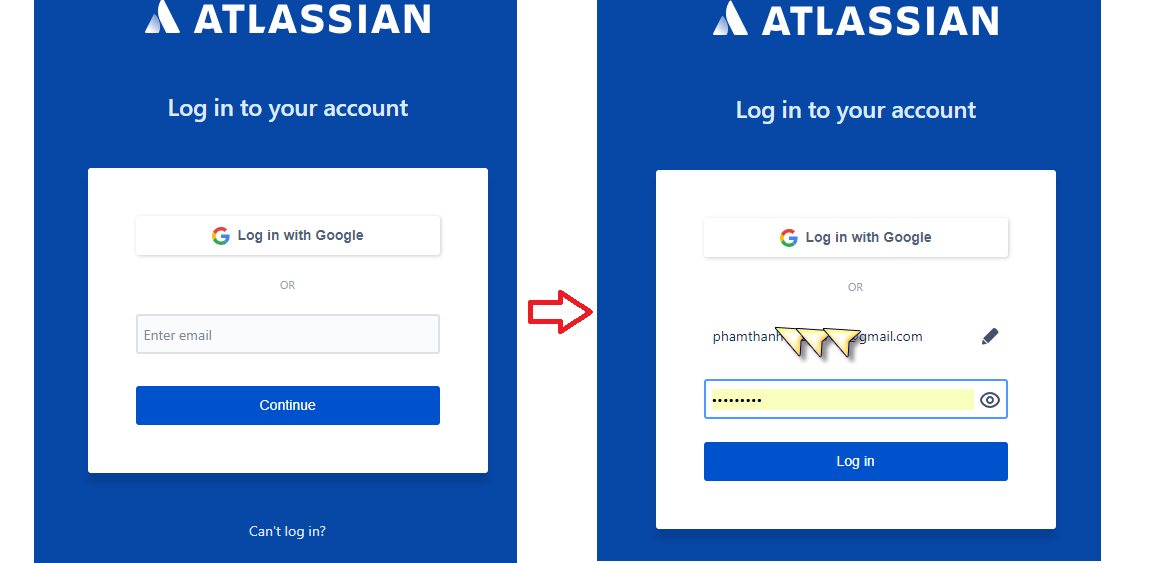
**CHƯƠNG 3: NỘI DUNG THỰC HIỆN**

## **3.1 Cài đặt**

Vào link <https://www.atlassian.com/> để sử dụng bản miễn phí

* 1. Cách sử dụng Jira

### 3.2.1 Đăng nhập vào Jira (Đối với Web)



Hình 3. 1 Đăng nhập

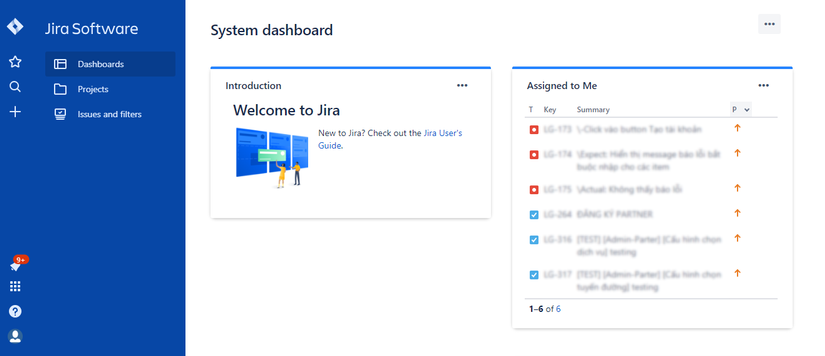
**a. Login bằng Tài khoản Google**

Bạn sẽ được yêu cầu login bằng tài khoản Gmail

* Nếu tài khoản đang được login trên trình duyệt thì sẽ được yêu cầu nhập mật khẩu…. (phần này tương tự như các ứng dụng khác - thông thường mọi người khá quen rồi)
* Nếu tài khoản chưa được login trên trình duyệt, thì sẽ được yêu cầu nhập thông tin Gmail để đăng nhập.

**b. Login bằng Email**

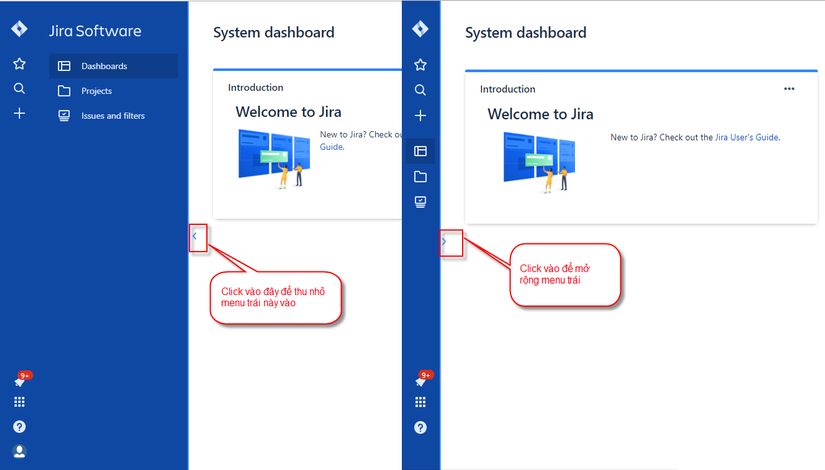
Với lựa chọn này ,thì bạn có thể sử dụng bất kỳ địa chỉ Email nào đã đăng ký với JIRA, (chú ý là bạn được mời vào dự án rồi)

* Bạn cần nhập vào địa chỉ email, và Click vào nút [Continue]
* Tiếp theo điền mật khẩu của tài khoản, Click vào [Log in] để đăng nhập.

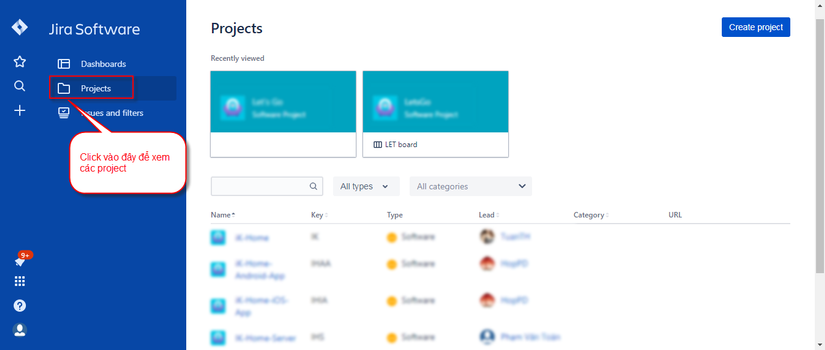
Hình 3. 2 Sau khi đăng nhập

### **3.2.2 Một số thao tác cơ bản**

**a. Mở rộng, thu hẹp Menu trái**  
Nhìn có vẻ đơn giản, nhưng khi mới dùng, chính bản thân mình cũng không biết làm sao để co nó lại, hoặc bật nó lên. Vì bình thường là nó ẩn nên không nhìn thấy, mà dùng màn hình nhỏ thì nhìn rất bất tiện. Cách thực hiện như sau:

* Đưa con chuột vào quãng giữa, sát mép lề của vùng menu trái, sẽ thấy mũi tên (< hoặc >)
* Click để đóng hoặc mở
* 

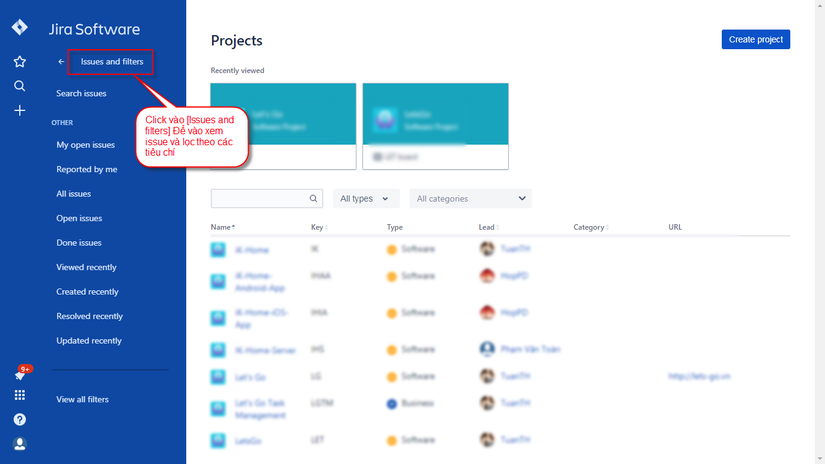
Hình 3. 3 Thu hẹp menu

**b. Xem Danh sách các dự án hiện tại**  
Từ Menu bên trái, click chọn [Projects] để mở màn hình Danh sách các dự án 

Hình 3. 4 Hiển thị danh sách các dự án

**c. Xem Danh sách các Issue, lọc tìm kiếm**

Từ Menu bên trái, click chọn [Issues and filters] để mở Submenu cho các lựa chọn

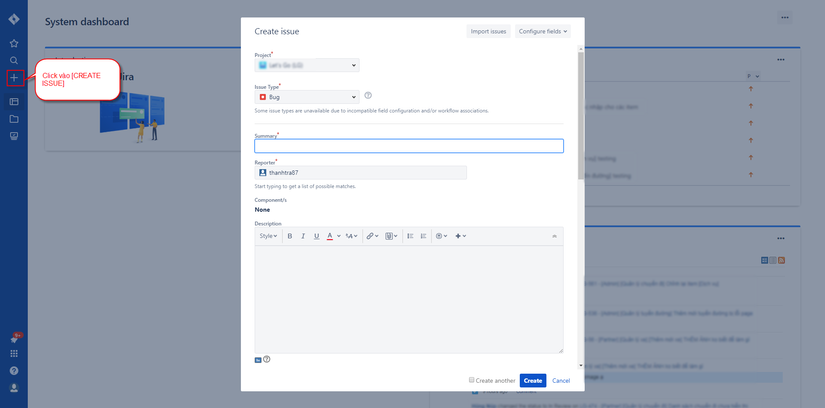
khác nhau

Hình 3. 5 Mở Submenu

* Search issues: Cho phép tìm kiếm các issue
* My open issues: Các issue đang được assign cho tài khoản đang login, chưa hoàn thành (To do, Inprogress)
* Reported by me: Các issue được tài khoản đang login tạo ra
* All issues: Tất cả các issue của tất cả các Tài khoản
* Open issues: Tất cả các issue đang ở trạng thái Open (Inreview/ Resolve, Inprogress, To do)
* Done issues: Tất cả các issue đã hoàn thành (trạng thái là Done/Close)
* Viewed recently: Tất cả các issue được xem gần đây
* Created recently: Tất cả các issue mới tạo gần đây
* Resolved recently: Tất cả các issue mới hoàn thành gần đây
* Update recently: Tất cả các issue mới có update gần đây

### **3.2.3 Tạo mới 1 issue**

* Click vào icon dấ (+) [Create Issue] để tạo mới 1 issue (hoặc có thể nhân phím tắt là chữ C)



Hình 3. 6 Tạo 1 issue mới

Note: Các mục có dấu (\*) là bắt buộc phải nhập Các mục này có thể cho hiển thị hoặc tát ở [Configure fields] Chi tiết các mục mình mô tả dưới đây:

**1. Project**\*: Chọn tên dự án (Issue tạo ra cho dự án nào). Với trường hợp bạn được add vào nhiều dự án, thì việc chọn dự án này rất quan trọng, nếu ko chú ý, có thể tạo nhầm dự án

**2. Issue Type**\*: Chọn loại issue Có thể là: Bug, Task, Epic….. (Loại issue này còn phụ thuộc vào việc Customize Jira của công ty đang sử dụng để phù hợp tình hình quản lý) Việc chọn loại issue sẽ thay đổi 1 vài item trên form, nên cần chú ý

**3. Summary**\*: Nội dung tóm tắt ngắn gọn mô tả cho Issue cần tạo

**4. Reporter**\*: Mặc định hệ thống sẽ nhận diện là User đang login

**5. Description**: Nhập mô tả cho issue, ví dụ:

* Môi trường
* Browser
* Device
* Steps thực hiện
* Kết quả mong muốn…

Ở đây có thể định dạng Font, mầu chữ, chèn file, ảnh, …. để dễ quan sát Phần mô tả này thì sẽ phụ thuộc vào loại issue, mà sẽ có mô tả như nào. Thông thường mình thấy description mô tả chi tiết thì việc giải quyết issue sẽ nhanh hơn

**6. Priority**: Mức độ ưu tiên. Tuy là mục option, nhưng nên đánh độ ưu tiên, để chú trọng giải quyết những issue quan trọng trước (Nếu không chọn thì mặc định đang để là Medium)

**7. Labels**: Chọn nhãn, hoặc đặt nhãn mới để có thể thống kê, tìm kiếm dễ dàng hơn. Ví dụ tên nhãn là tên các chức năng, nghiệp vụ

**8. Original Estimate**: Thời gian dự kiến hoành thành

**9. Remaining Estimate**: Thời gian còn lại để hoàn thành

**10. Environment**: Môi trường sử dụng là gì, hãy điền vào đây

**11. Attachment**: Gắn file đính kèm

**12. Due date**: Dự kiến ngày phải hoàn thành

**13. Linked Issues, Issue**: Liên kết tới issue nào đó đã có

**14. Assignee**: Người sẽ giải quyết issue

**15. Epic link**: Liên kết tới Epic tài liệu nào

Ở cuối form nếu tick chọn vào checkbox [Create another], thì sau khi create xong issue, sẽ giữ lại form trên màn hình, cho phép tạo issue tiếp theo

### **3.2.4. Thao tác với 1 issue**

Với developer, thì việc thao tác với 1 issue sẽ nhiều không kém 1 Tester (hay QA)

* Mở Bug/Task thực hiện những
* Chọn các mục sau



Hình 3. 7 Thao tác với 1 issue

**1. Edit**: Sửa lại issue. Cho phép sửa lại hầu hết các mục như khi tạo mới issue

**2. Comment**: Add thêm comment

* Thường dùng khi thực hiện 1 task/bug, sẽ comment qua lại để trao đổi, hoặc chỉ đơn giản là comment fix bug, hoặc thực hiện task xong chưa, thực hiện được deploy vòa bản nào
* Có thể gắn thêm user khác vào (sẽ bắn mail. push notify báo cho user đó) bằng cách viết: @[+tên user]

**3. Assign**: Đổi lại Assignee

Đôi khi trong dự án, có 1 task, bug... assgin cho 1 ai đó (hoặc cho mình) nhưng là do assign nhầm, hoặc muốn đổi lại sang cho member khác làm, thì ta có thể sửa lại mục này

**4. In Progress**: Chuyển trạng thái sang [In Progess] cho bug/ task khi bug/task đang trong giai đoạn thực hiện, nhưng chưa hoàn thành

**5. Workflow**: có 2 lựa chọn [In review] và [Done]

[In review] thường dùng để đánh dấu là task/ bug đã hoàn thành giai đoạn thực hiện, còn chờ kiểm tra lại

[Done] thường dùng để đánh dấu task/bug đã hoàn tất kiểm tra note: mục này còn tùy thuộc vào từng phiên bản Jira, hoặc tùy thuộc vào việc customize Jira, nên có thể sẽ khác

**6. Share**: Chia sẻ với user khác

**7. Export**: Xuất file ra format khác (thường là Excel, hoặc CSV)

**8. Option**: Thêm rất nhiều lựa chọn thao tác với issue

* Log work: Log từng mốc thời gian thực hiện của Task/Bug
* Attach files: Đính kèm thêm file
* Create sub-task: Tạo SubTask cho Issue hiện tại
* Convert to sub-task: Chuyển issue hiện tại thành subtask của 1 issue nào đó...

### **3.2.5. Tìm kiếm, lọc issue**

Mục này đôi khi hay bị bỏ qua, nhưng nó rất hữu ích. Nhất là khi bạn cần tìm issue nào đó, hay là list những issue nào đó mà danh sách thì quá dài,...

* Vào Menu / Issues and filter / All issues
* Nếu danh sách issues đang hiển thị dạng 1 cột List, 1 cột Detail thì nên chuyển sang chế độ chỉ có List Issue cho dễ quan sát, bằng cách: Click vào [Advanced search] góc trên bên phải màn hình

Hình 3. 8 Tìm kiếm, lọc issue

Các mục hay sử dụng

**1. Project**: Chọn tên dự án

**2. Type**: Chọn kiểu issue (Bug, Task, Epic, SubTask ….)

**3. Status**: Trạng thái của Issue (To do, In Progress, In review / Resolve, Done / Close, Cancel…)

**4. Assignee**: Tên user được phân giải quyết issue

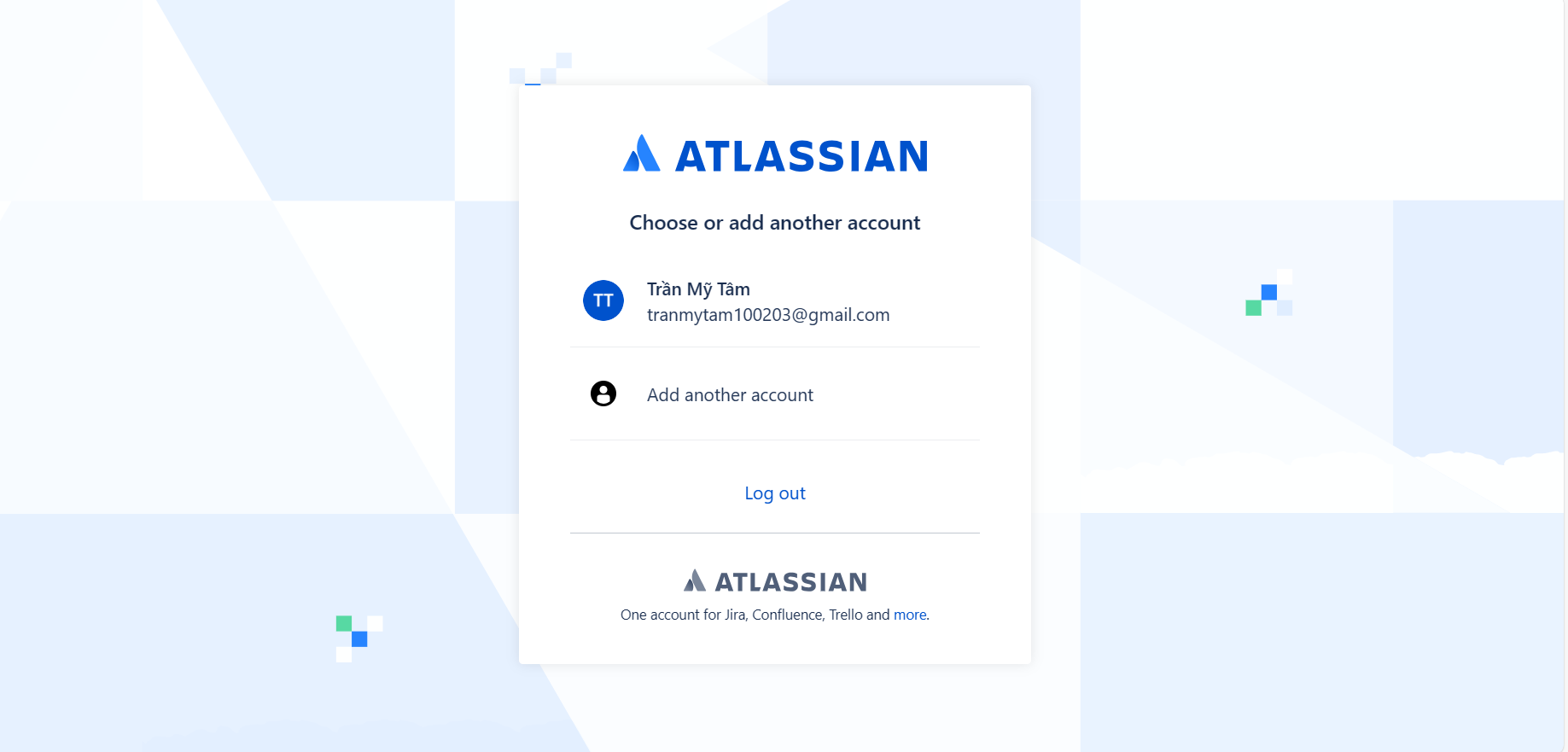
**5. Textbox** (Contains text): Nhập từ khóa có chứa trong issue cần tìm

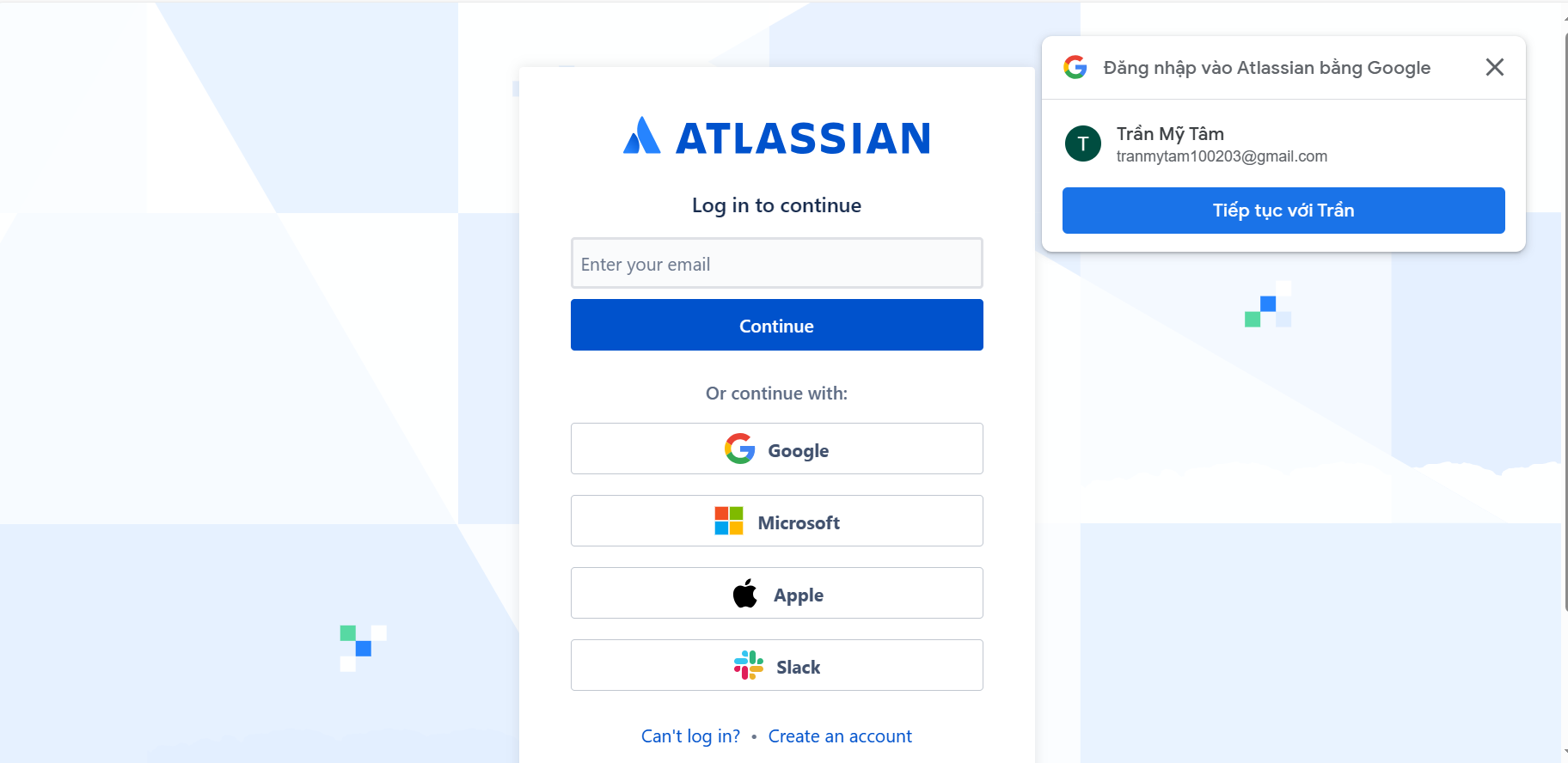
**6. More**: Thêm nhiều tiện ích hay được dùng để tìm kiếm hơn như:

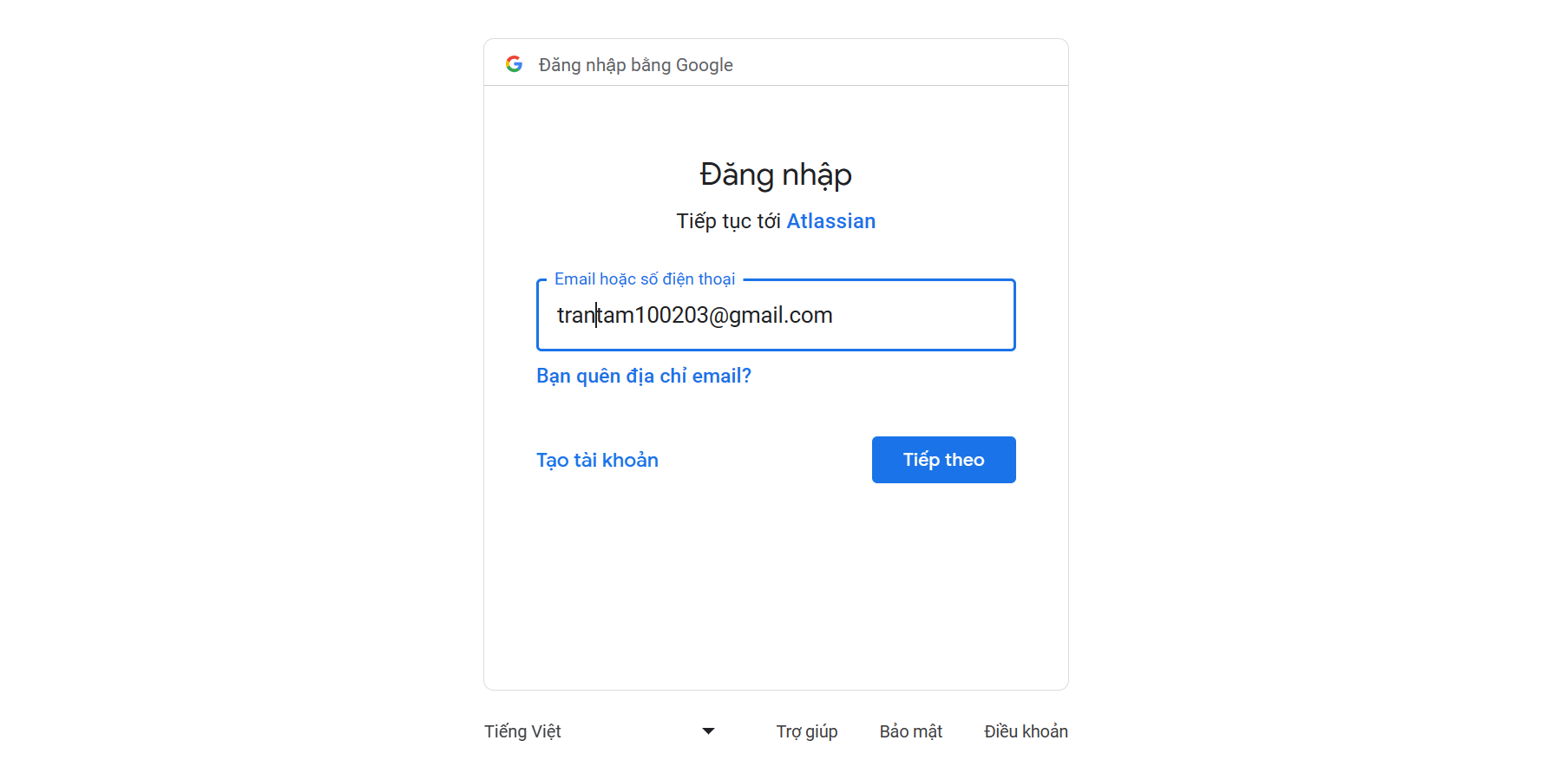
* Updated Date
* Created Date
* Reporter
* Label
* Description
* Priority...

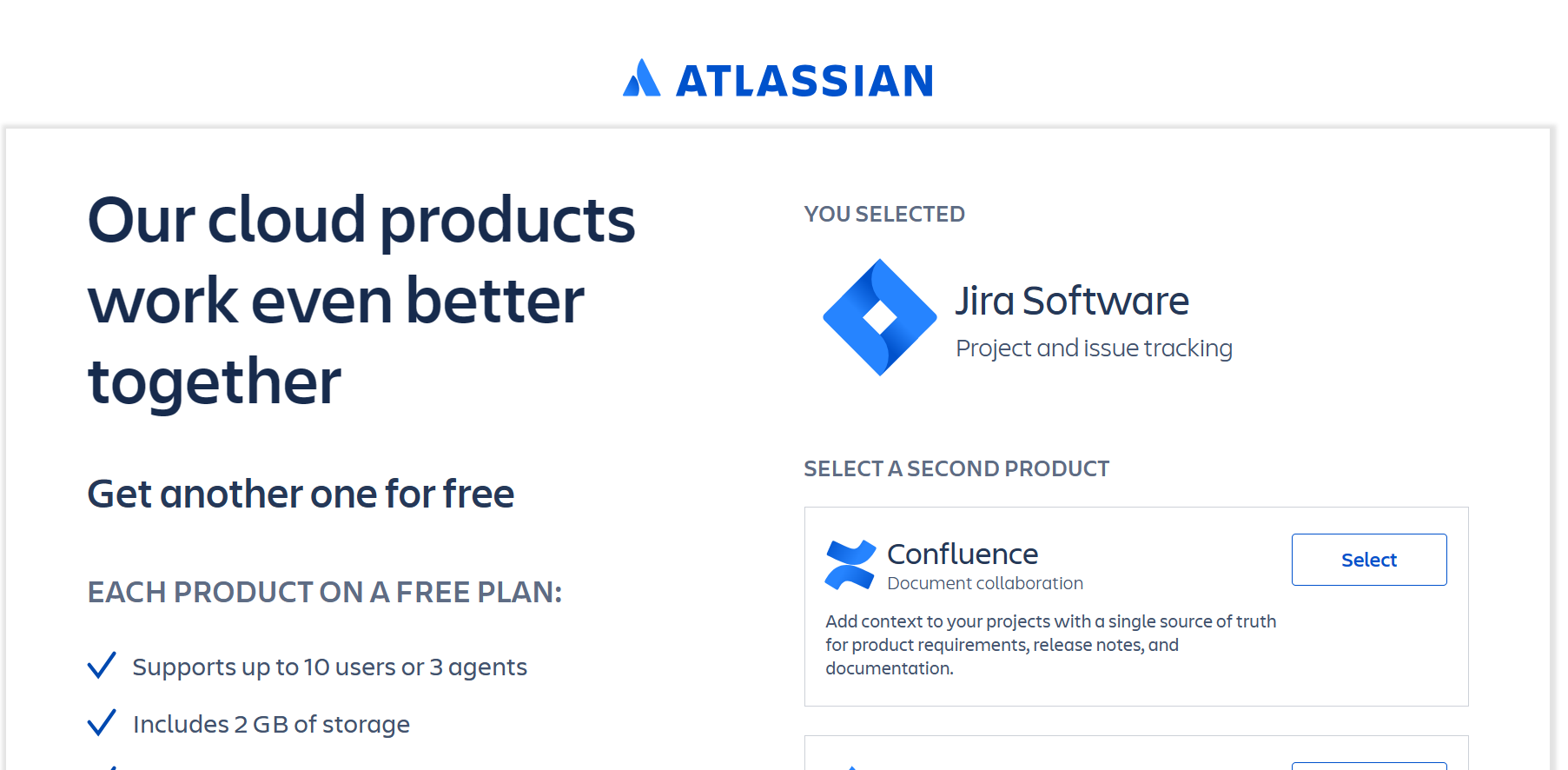
Ngoài ra chúng ta có thể customize việc hiển thị các cột trên list danh sách, bằng cách vào [Columns] để lựa chọn .

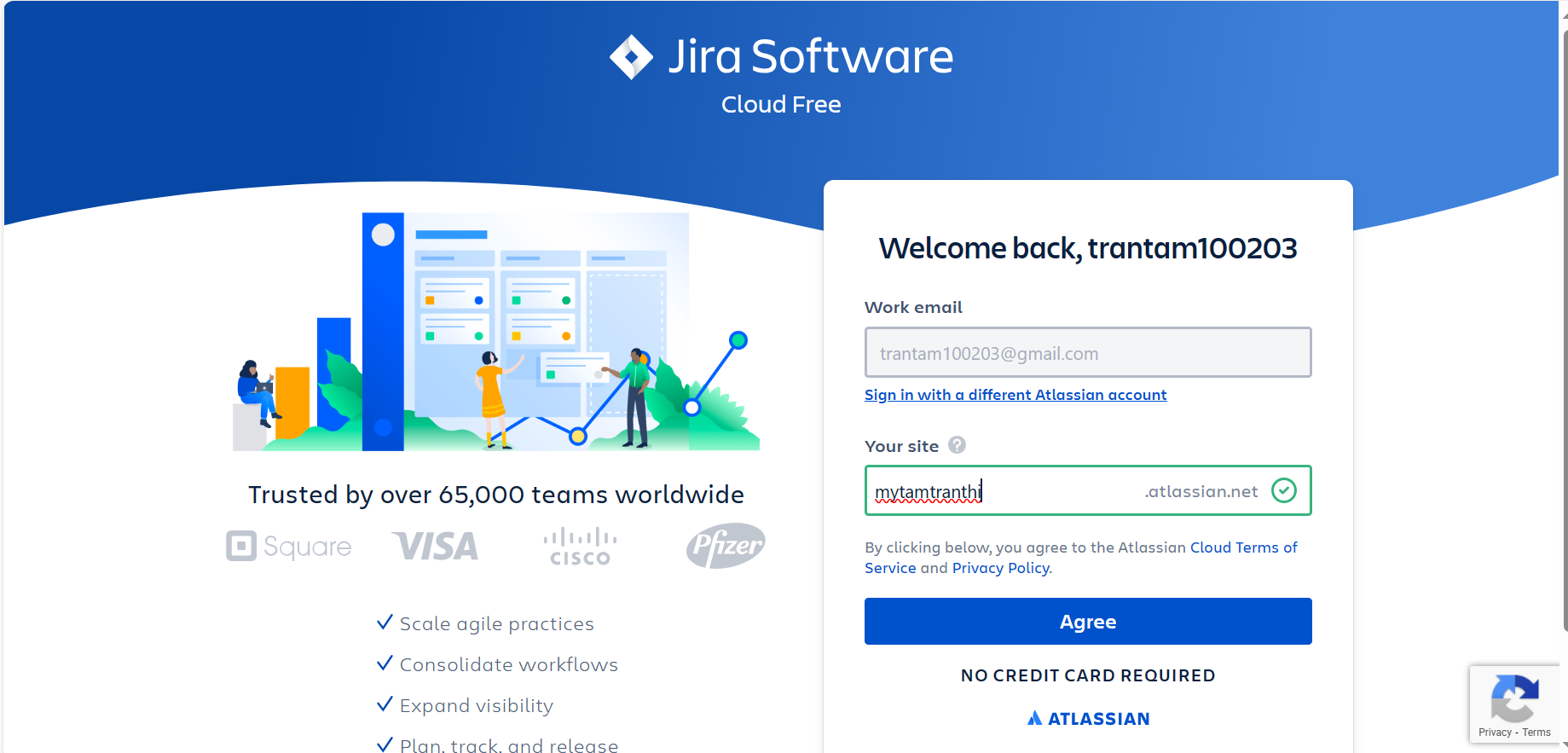
* 1. Áp dụng vào dự án cụ thể (Đồ án 2)
     1. Tạo tài khoản miễn phí trên Atlassian

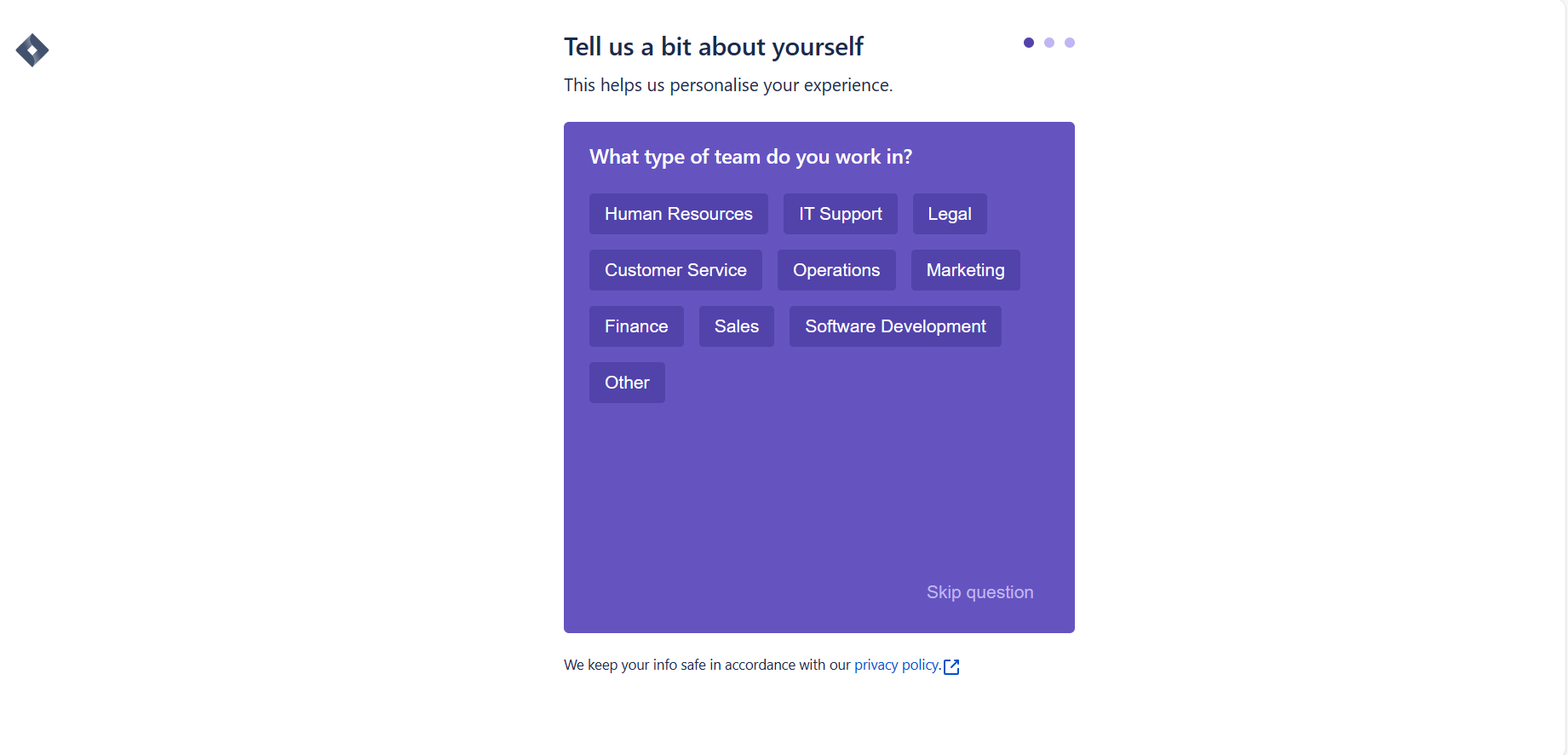


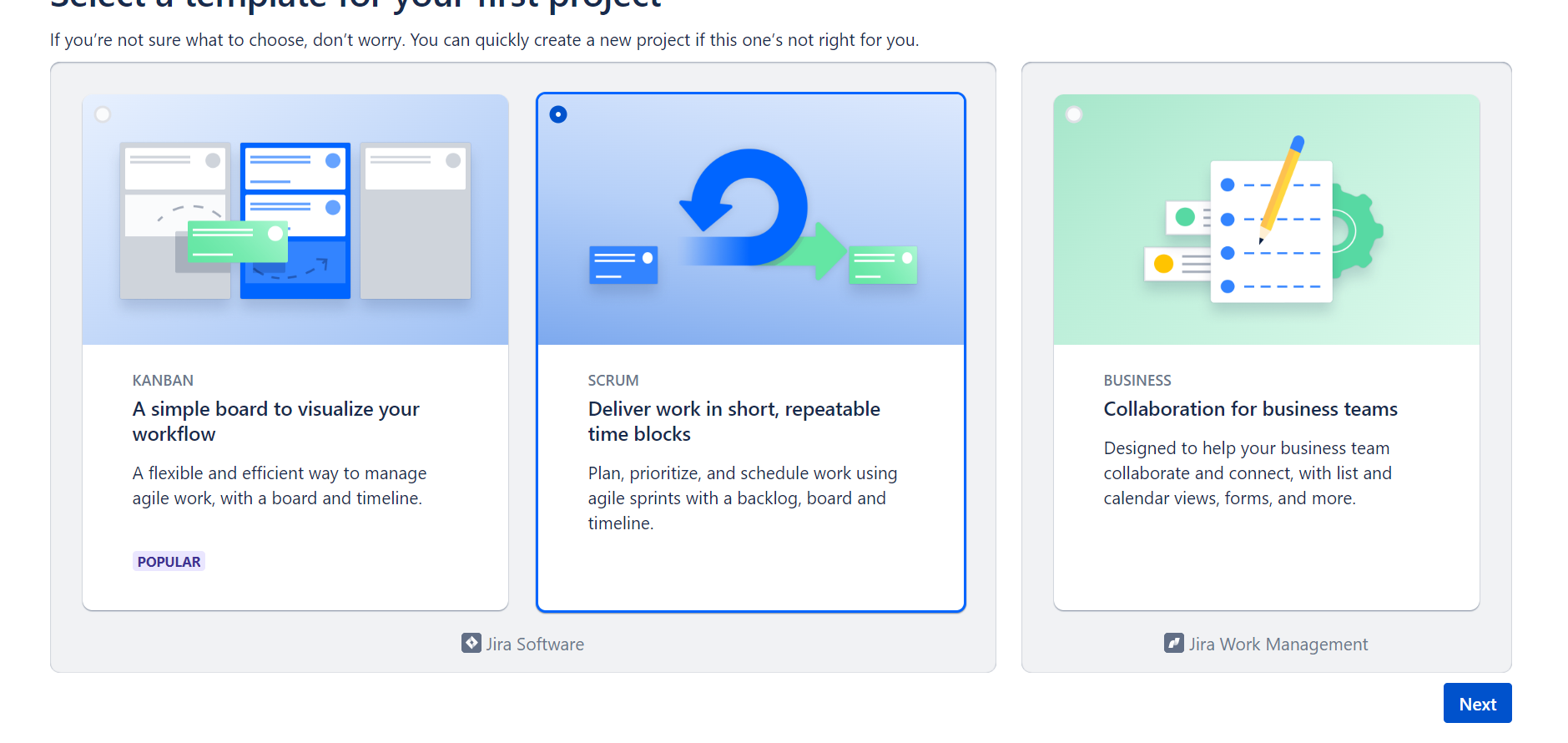






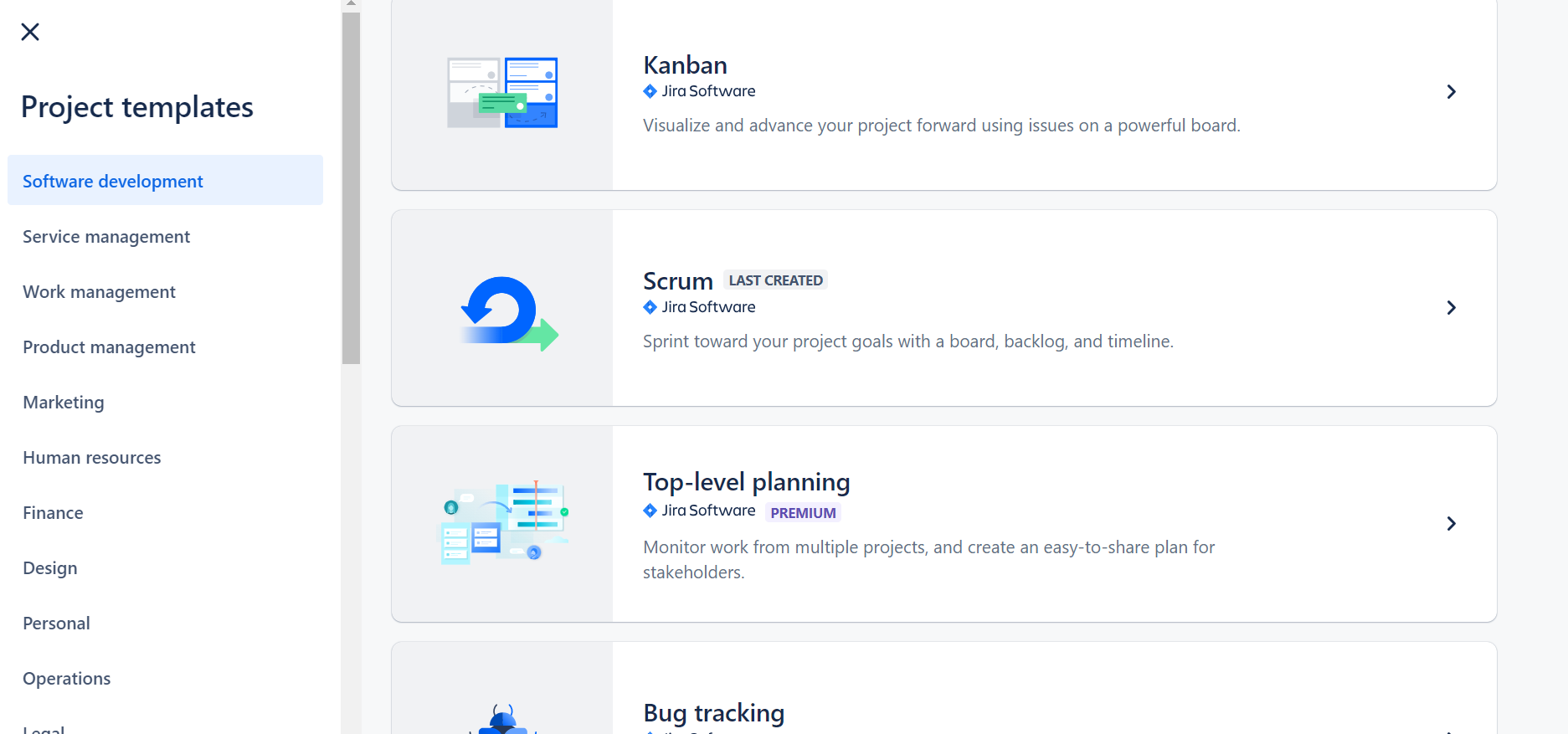




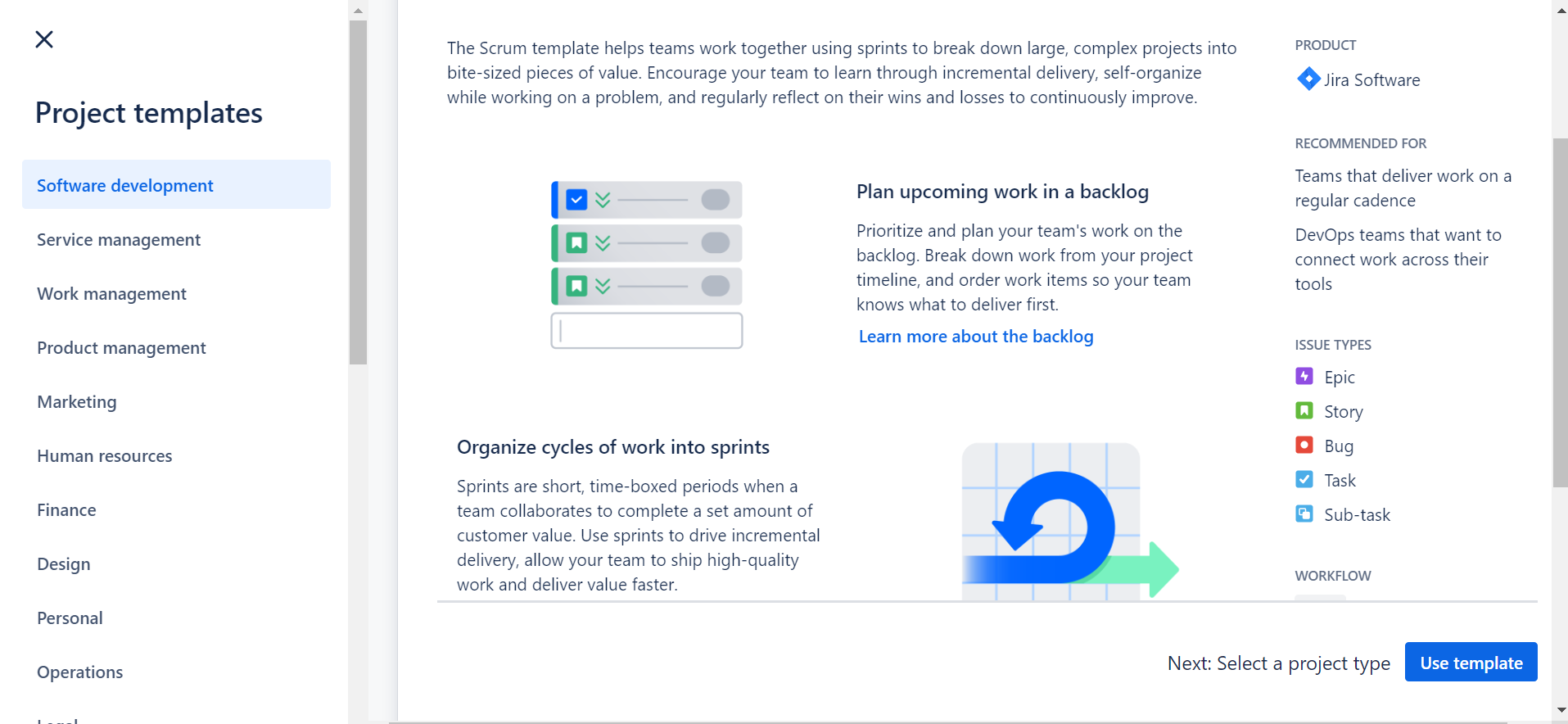




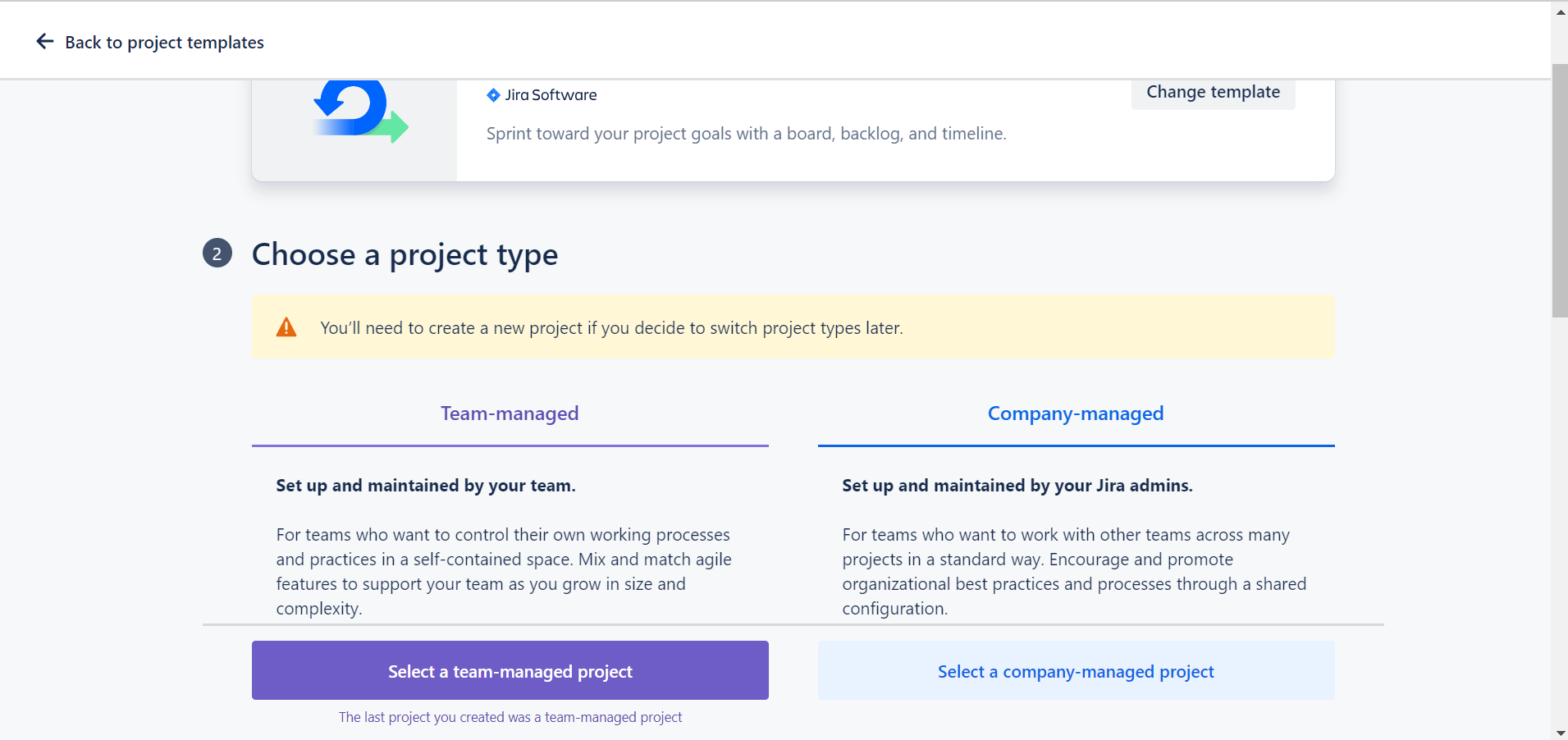
* Chọn Crum

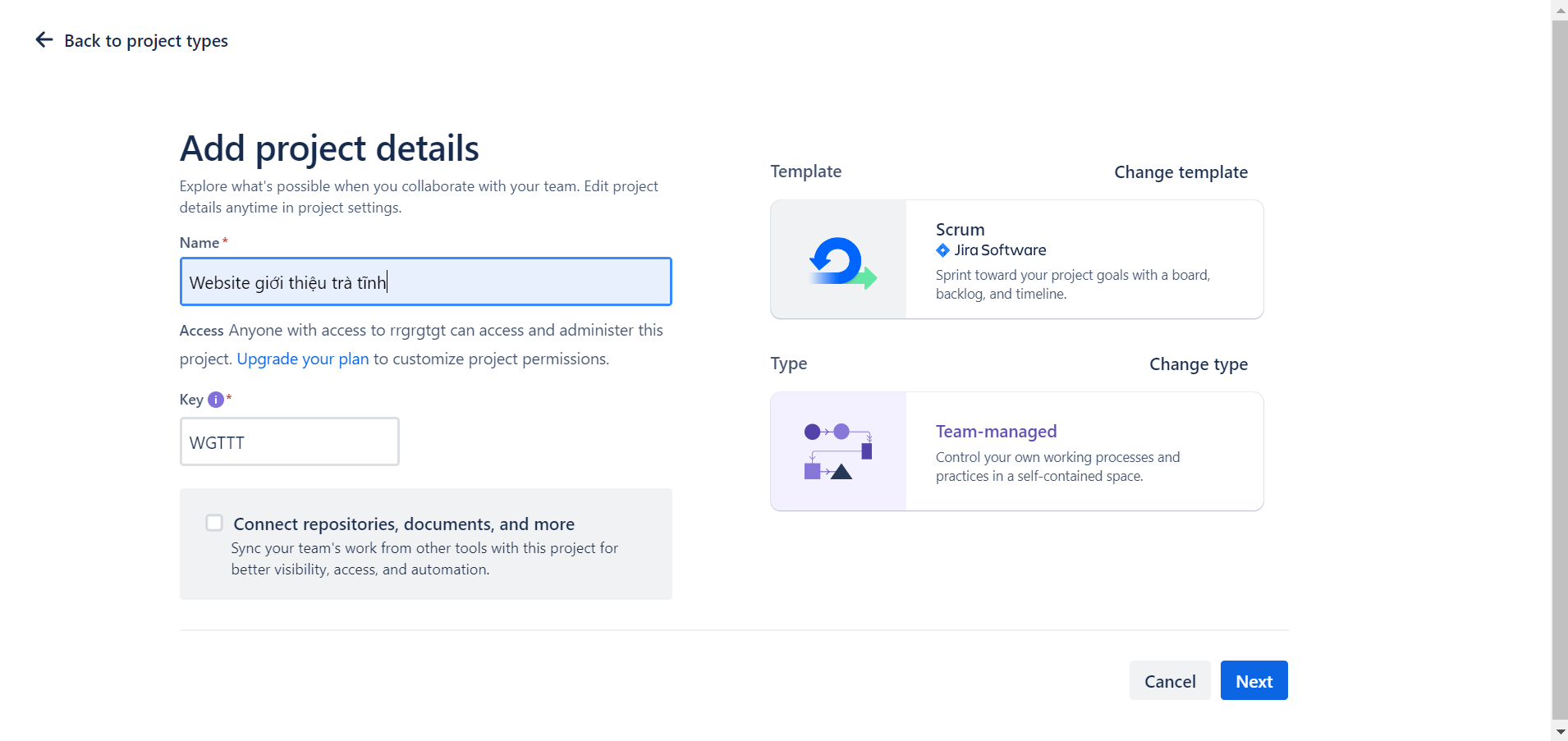


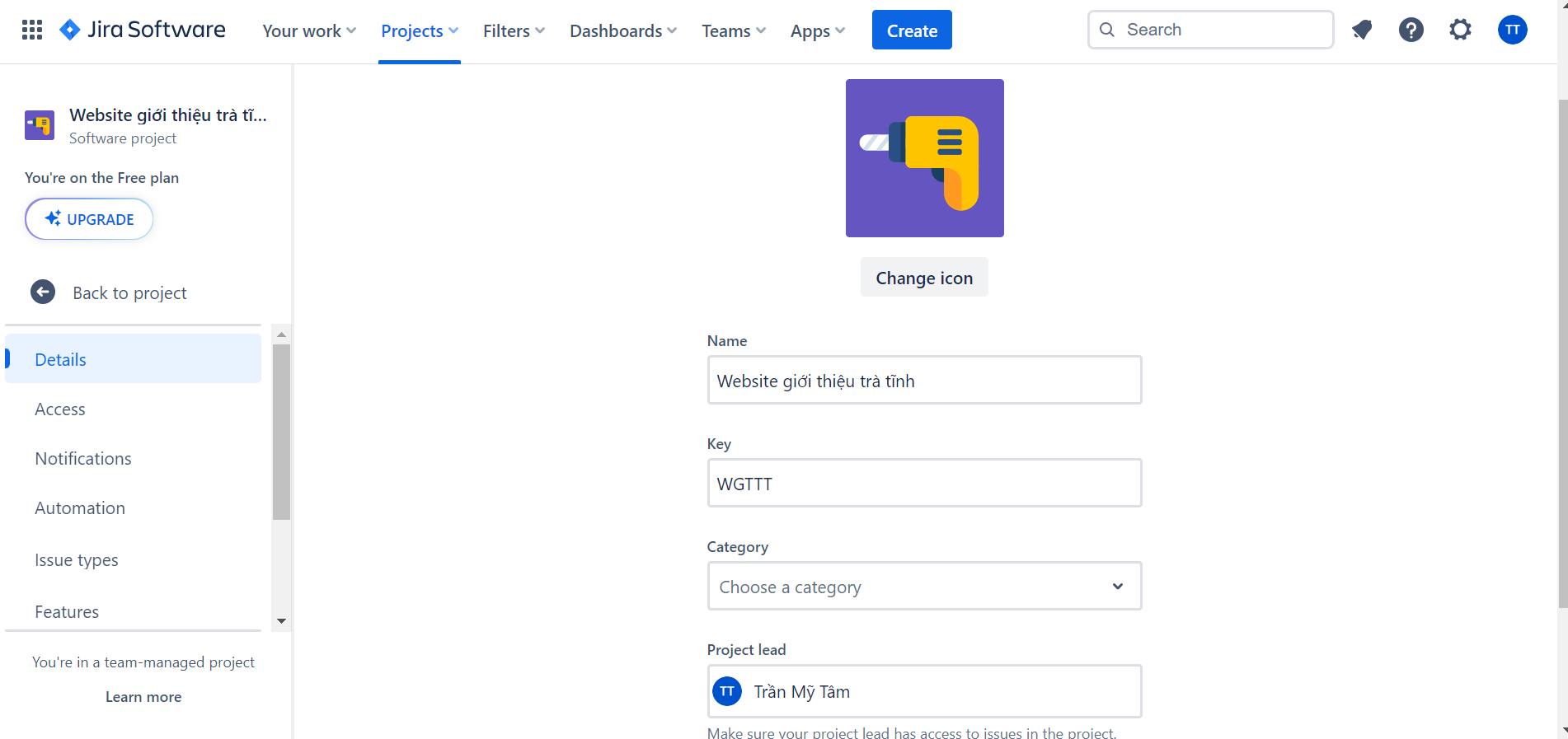
* Chọn use template

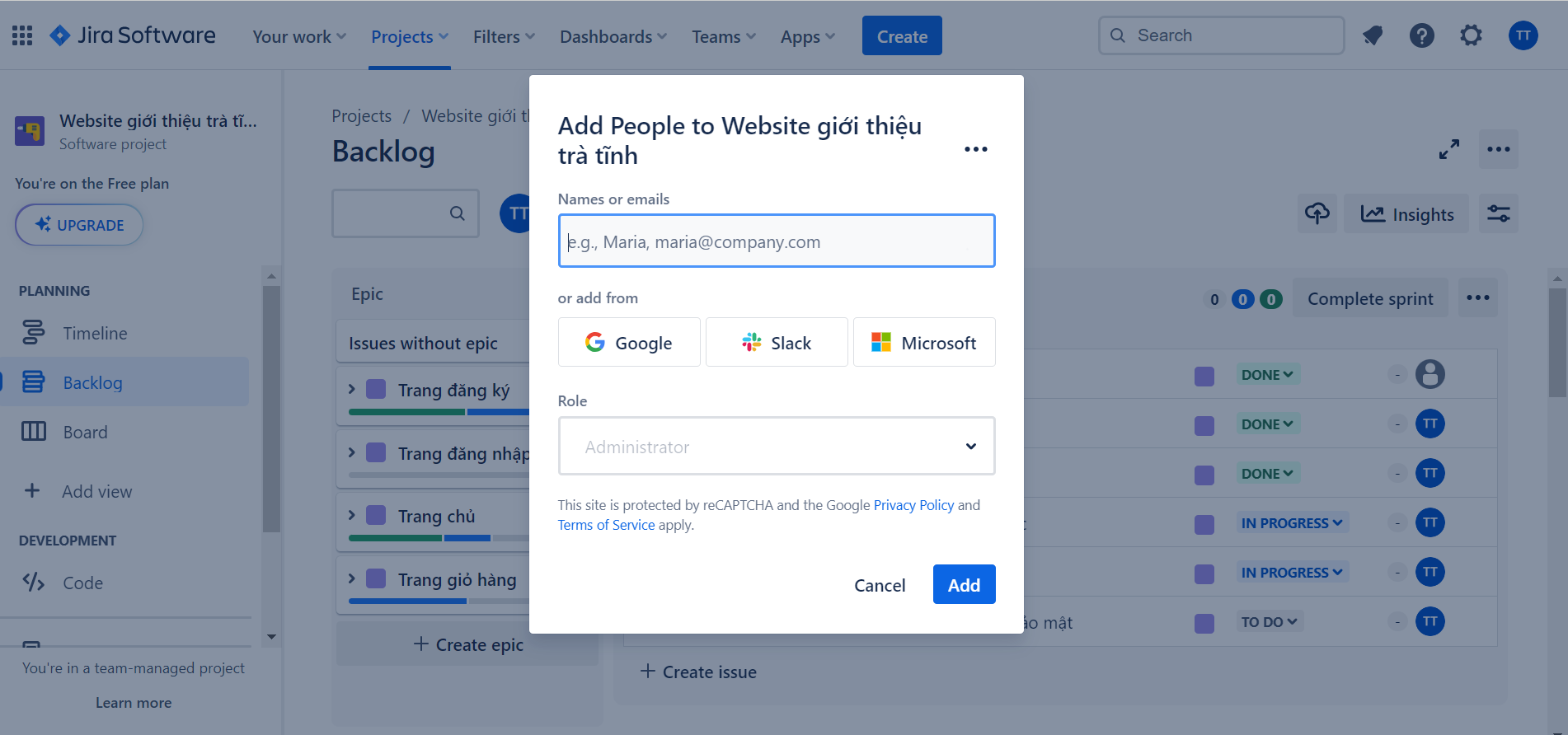


* Chọn select a team manage

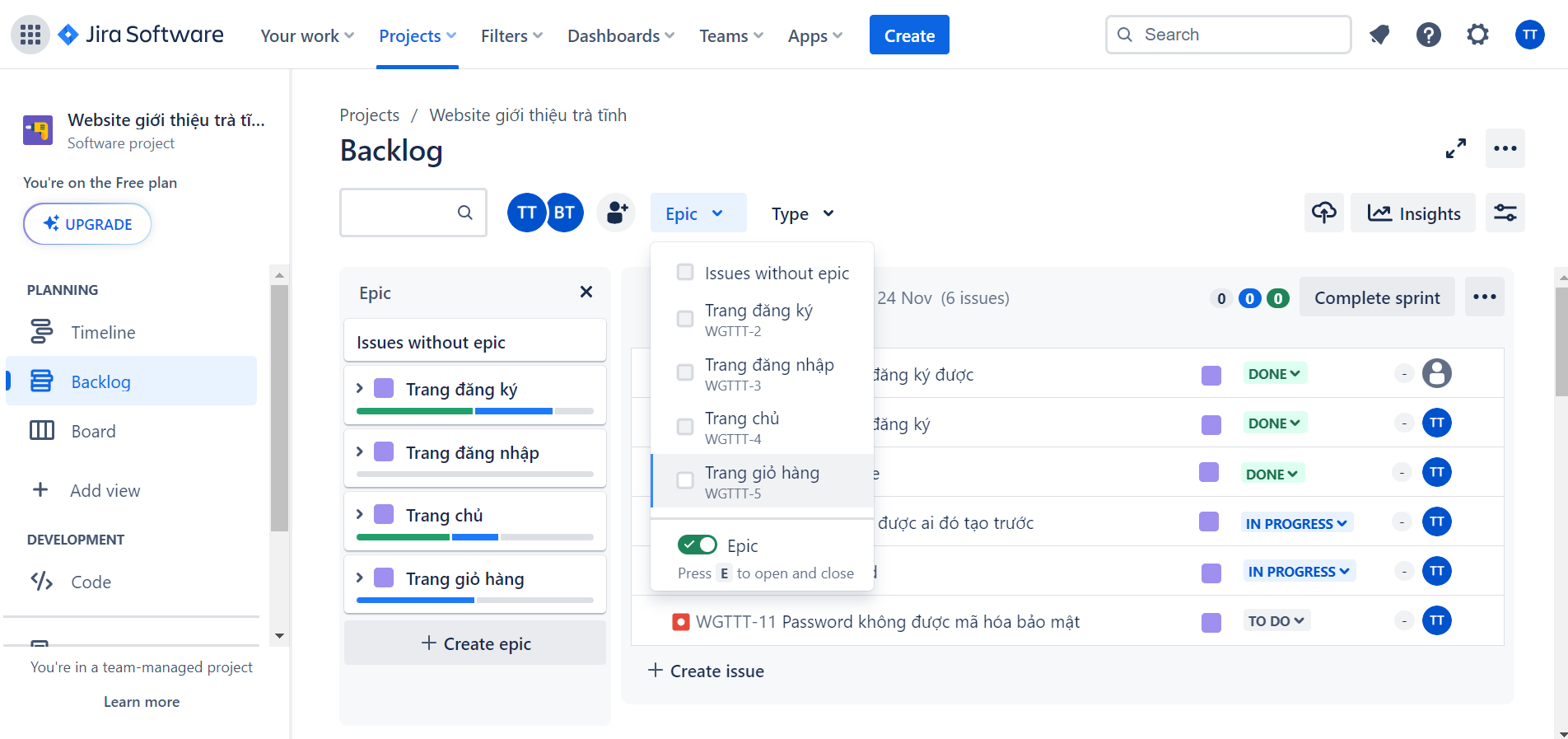


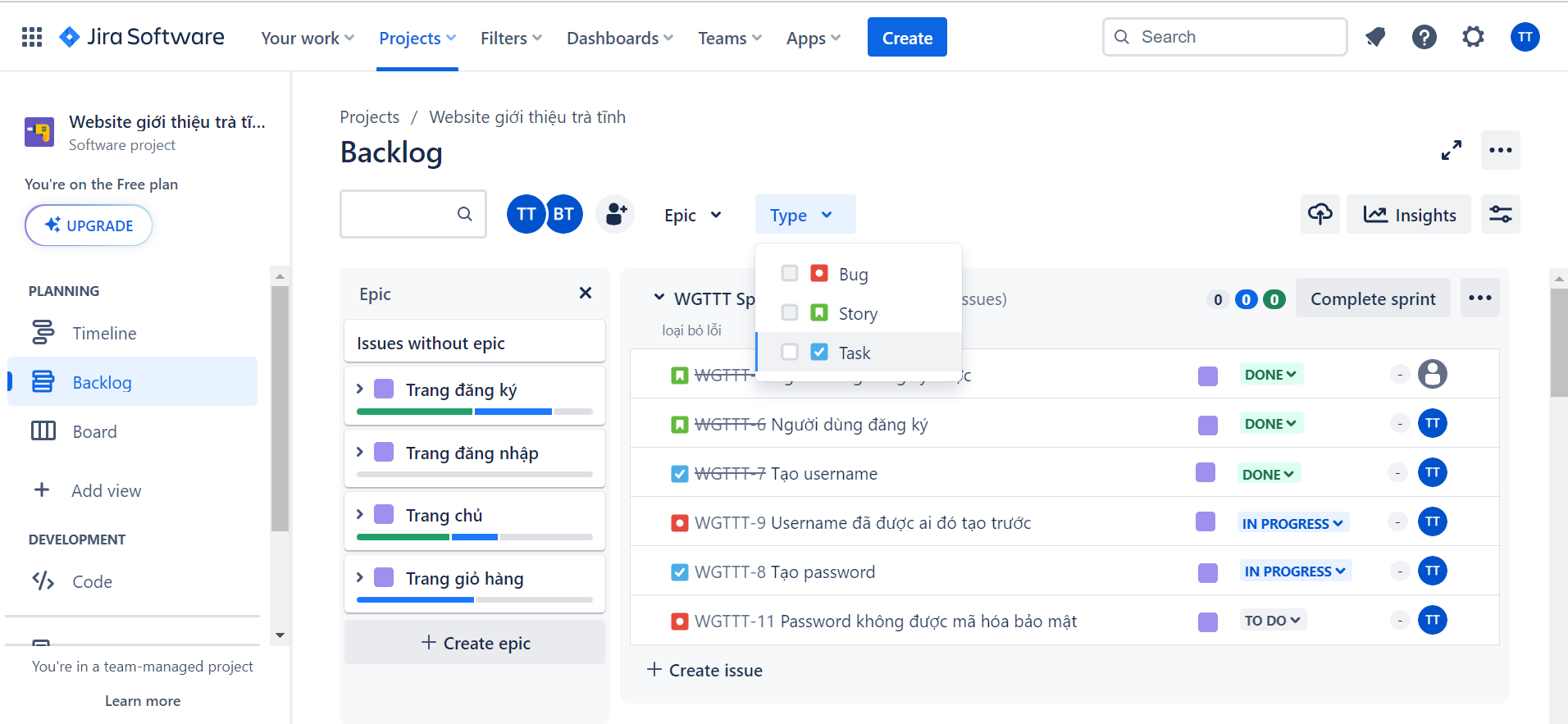


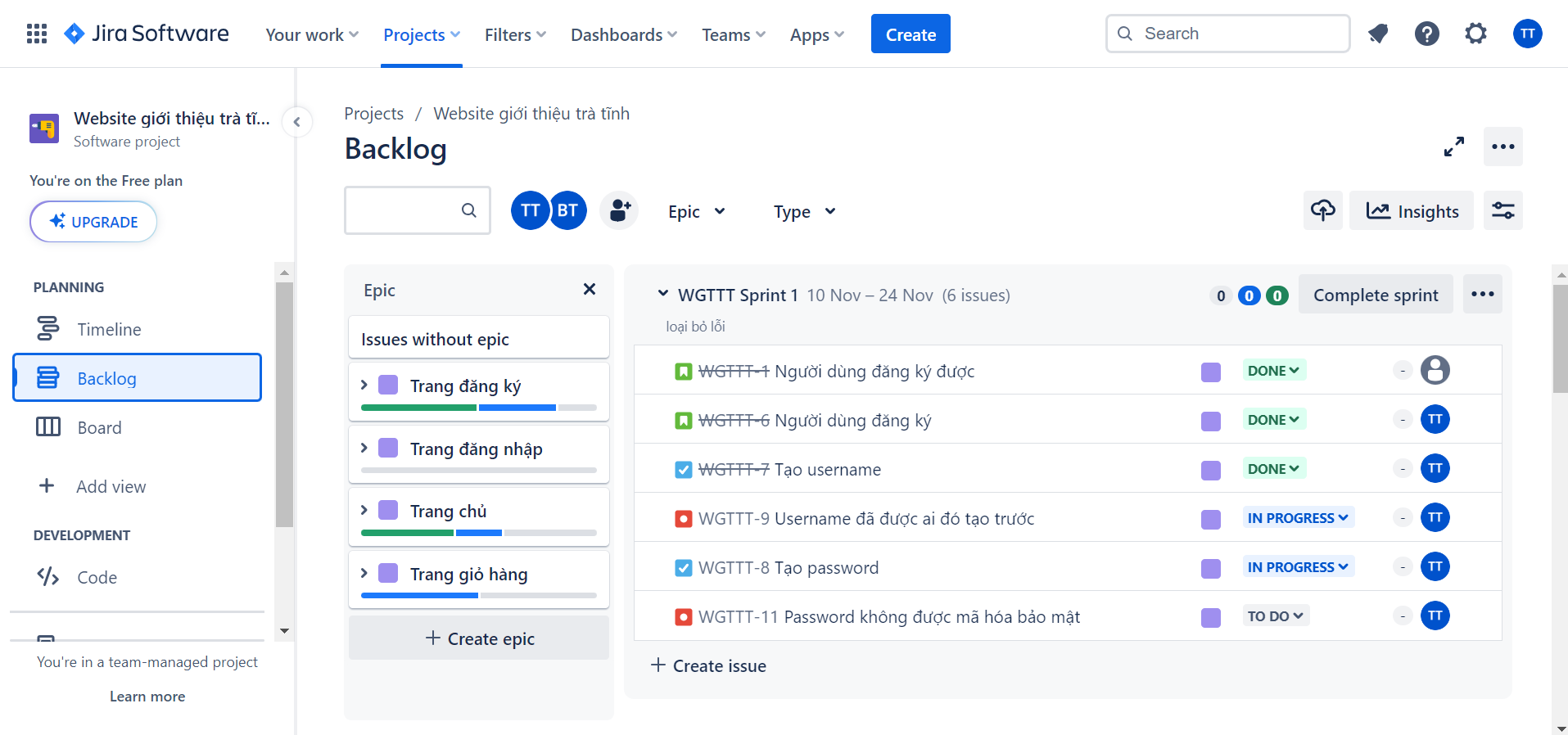


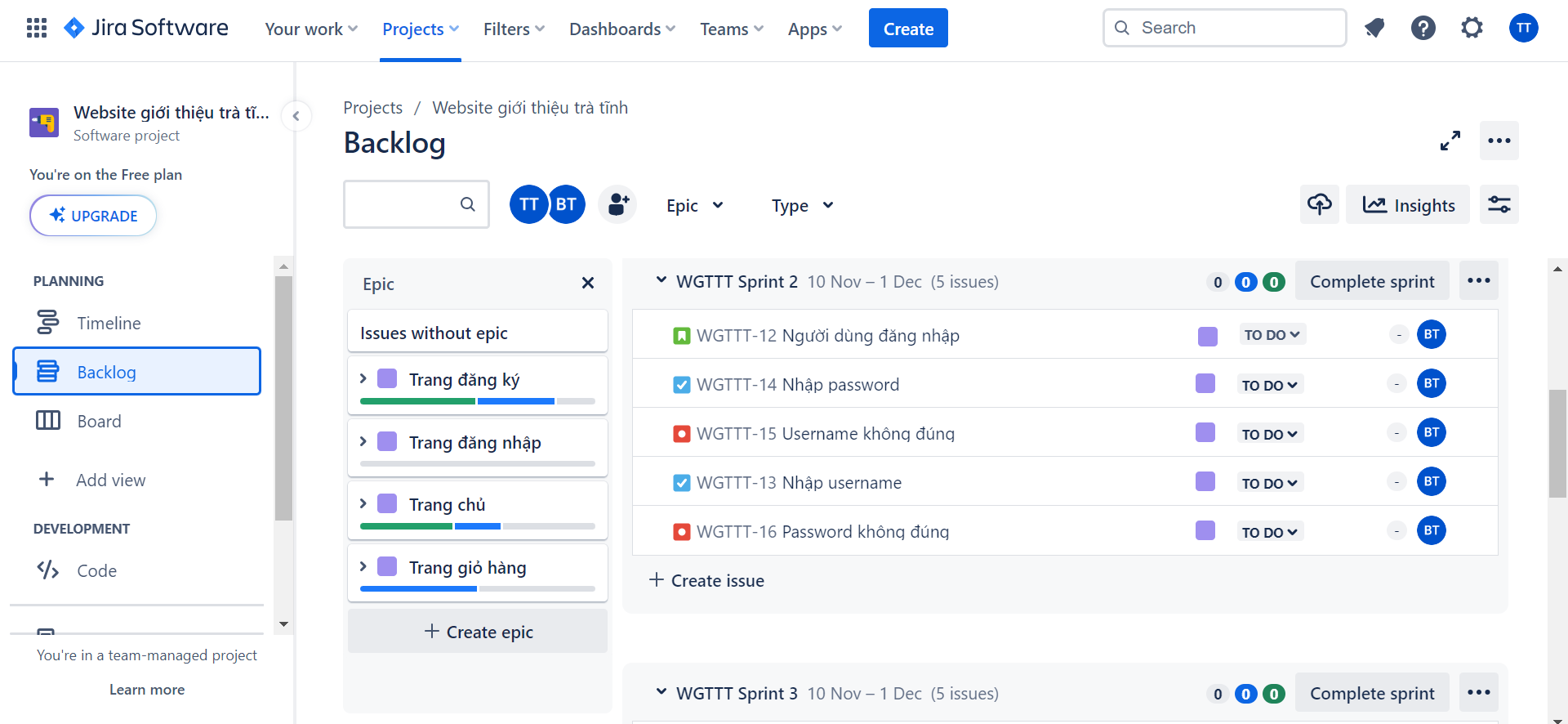


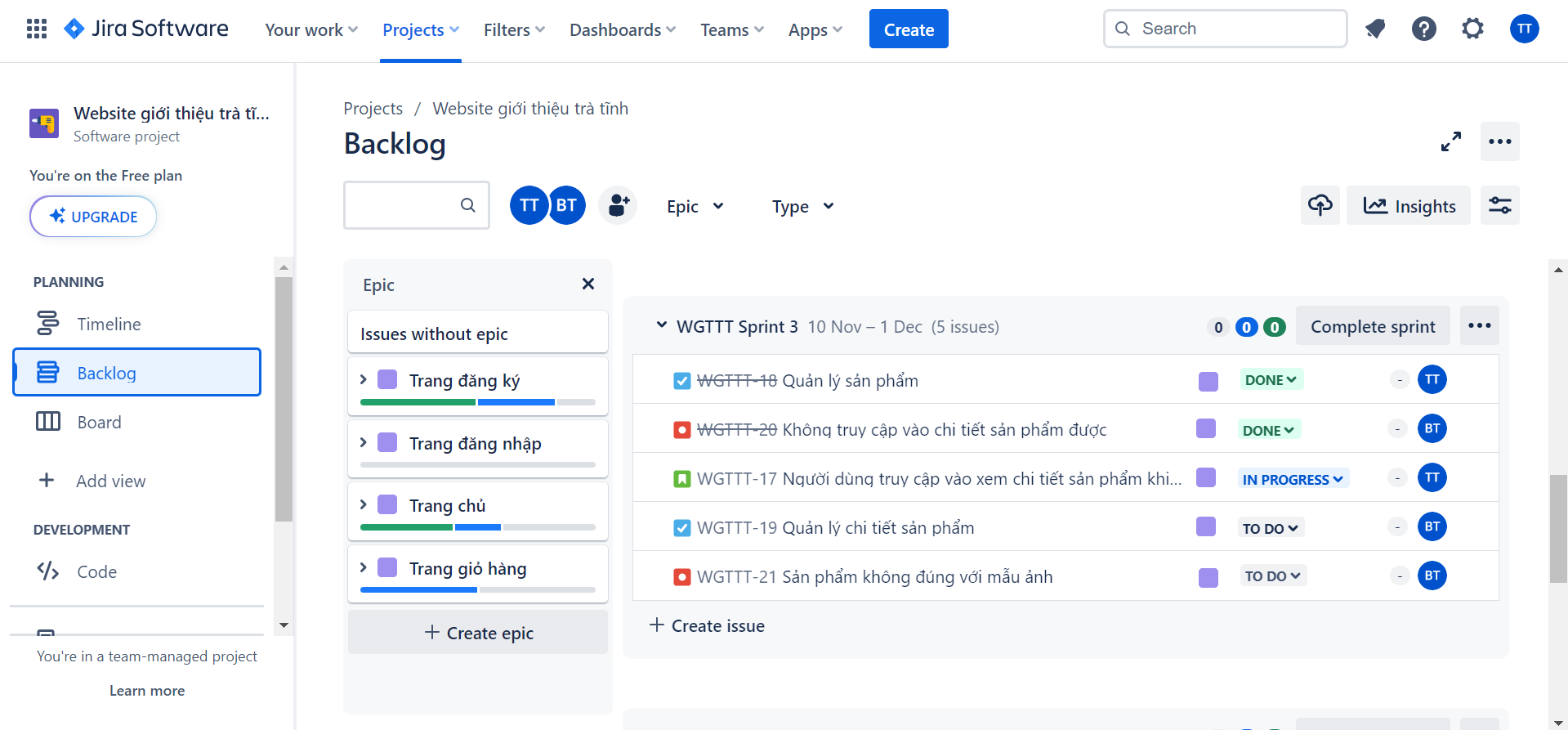
**3.3.2 Áp dụng vào đồ án 2- Xây dựng website giới thiệu trà**

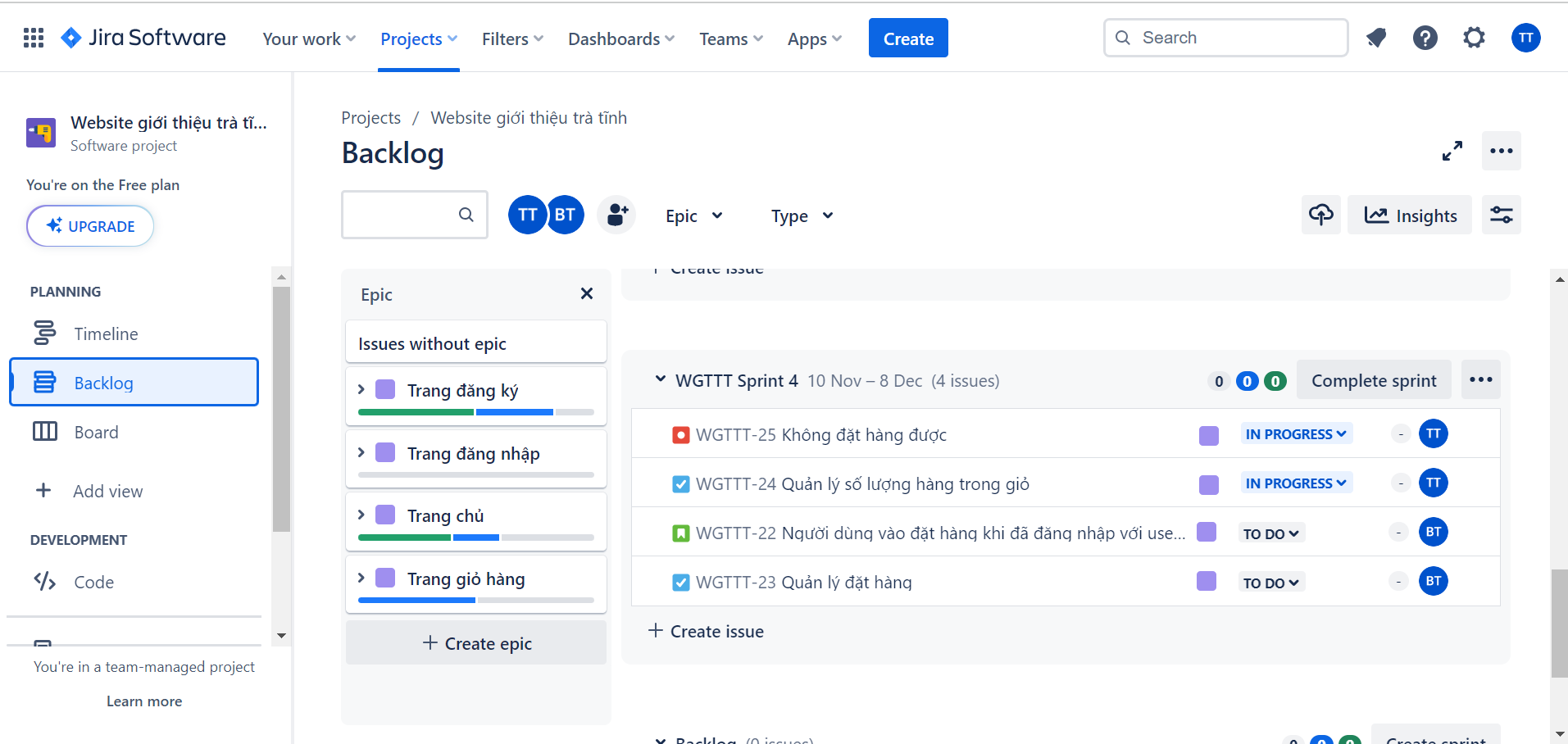


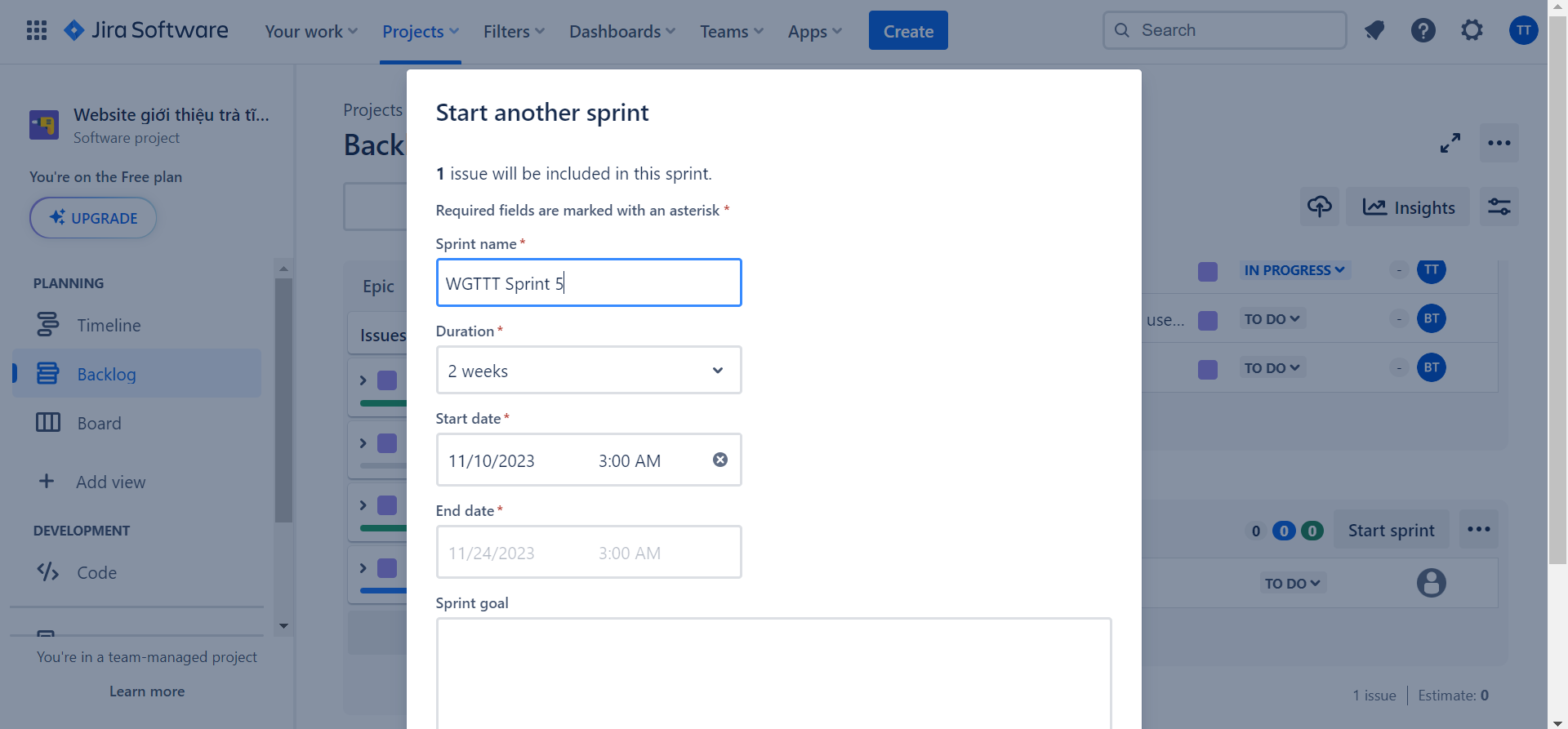


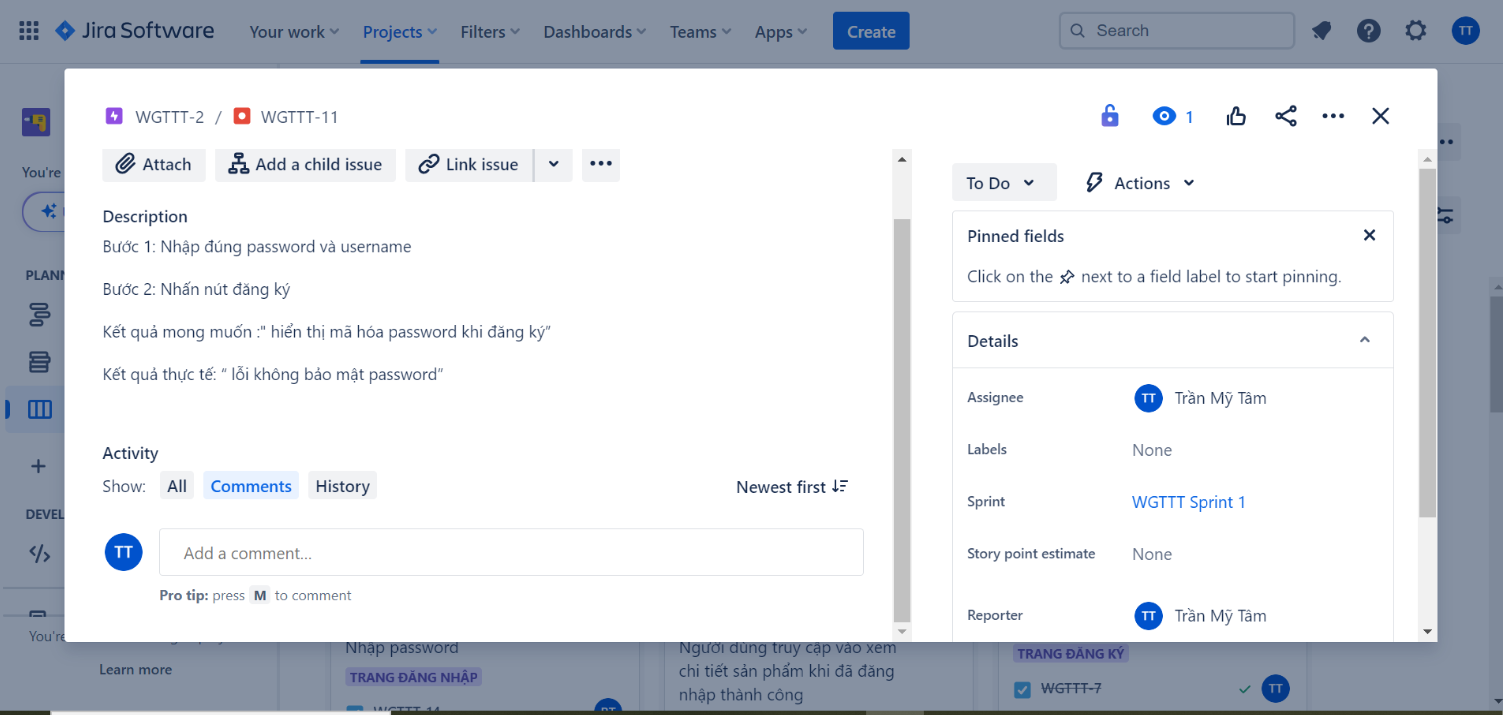


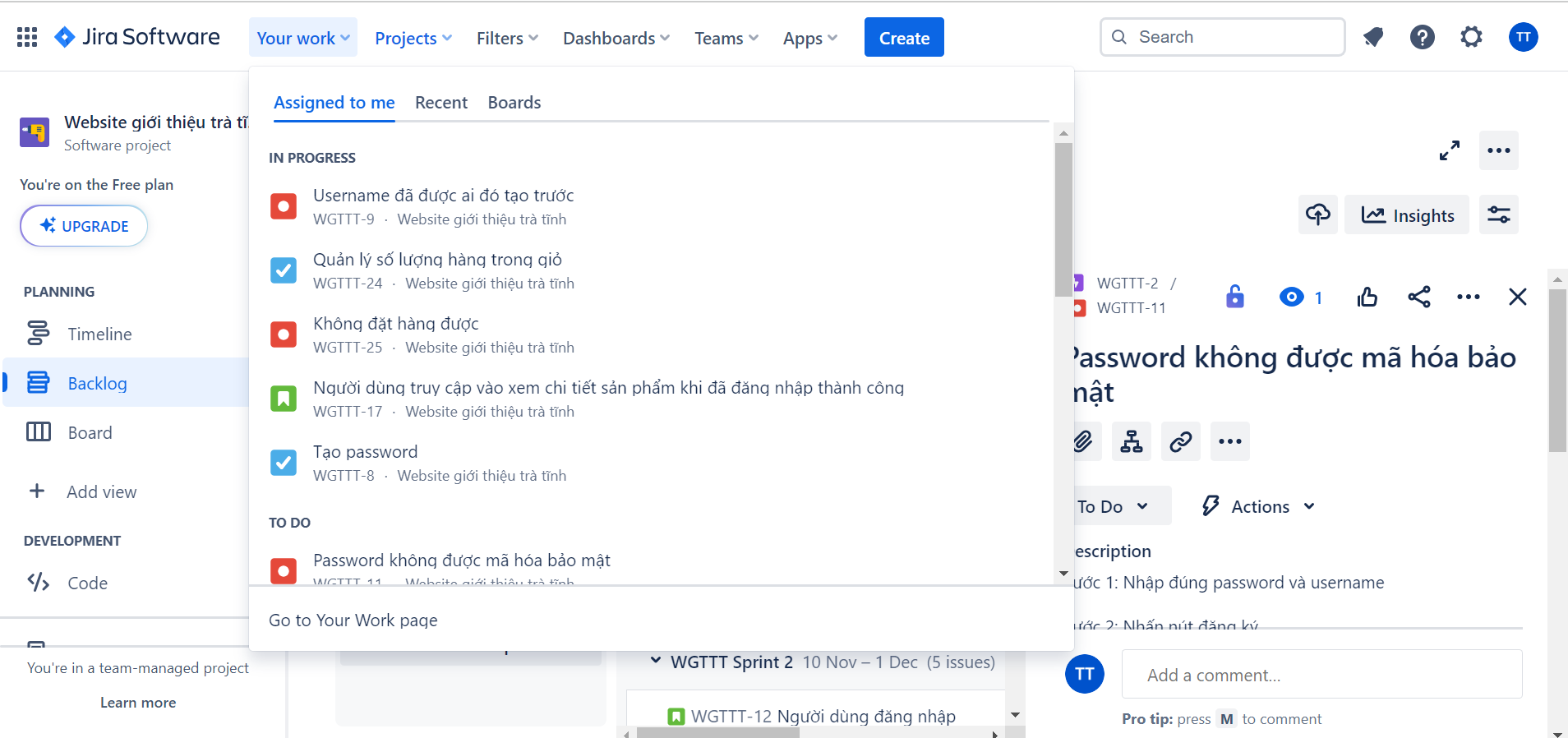


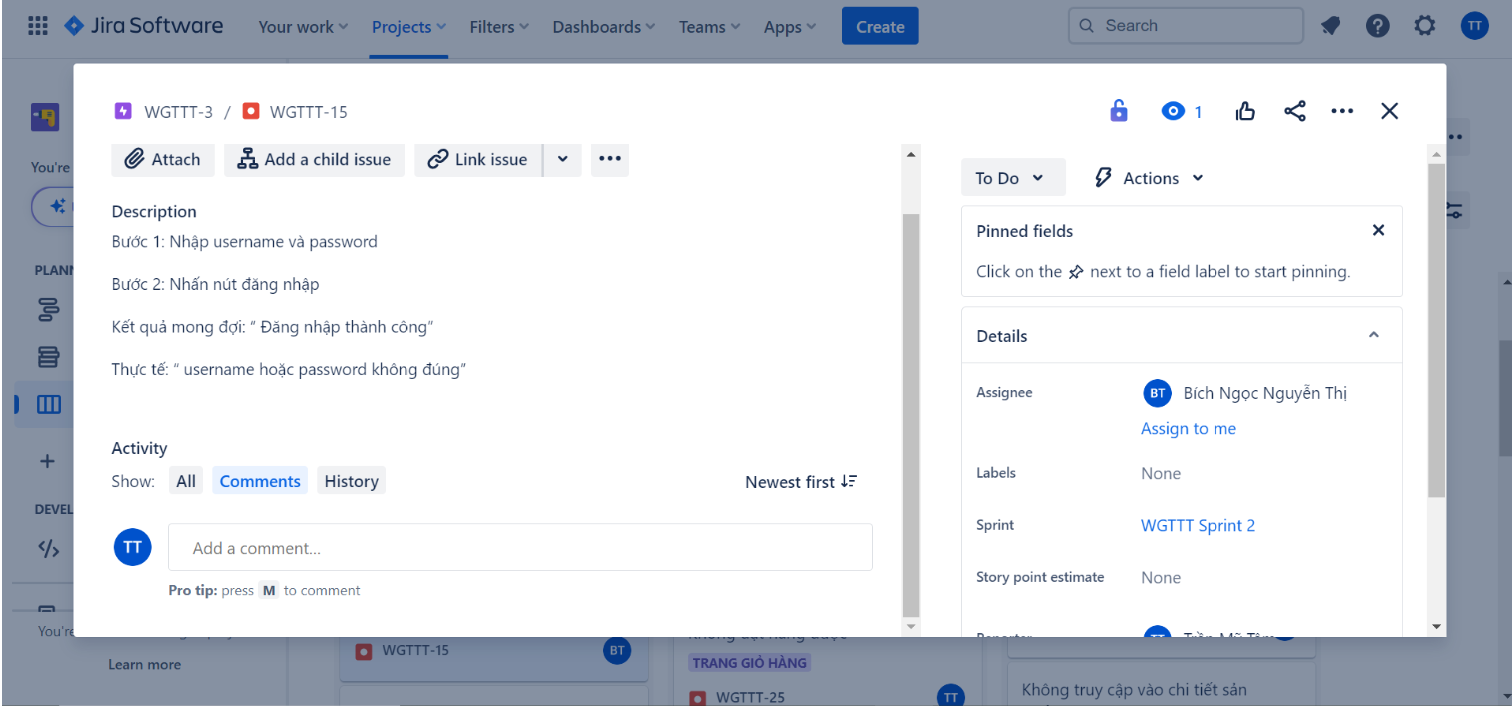


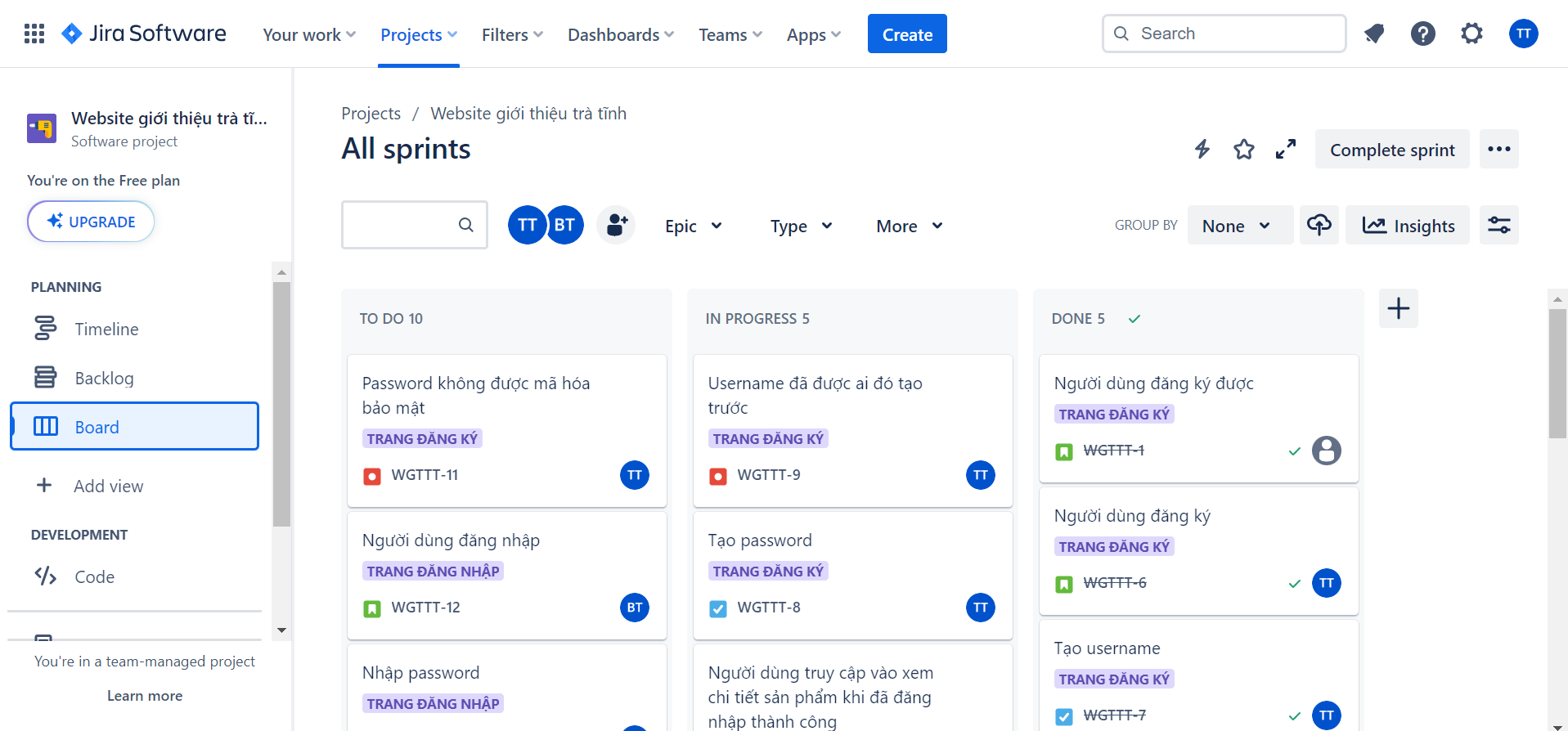


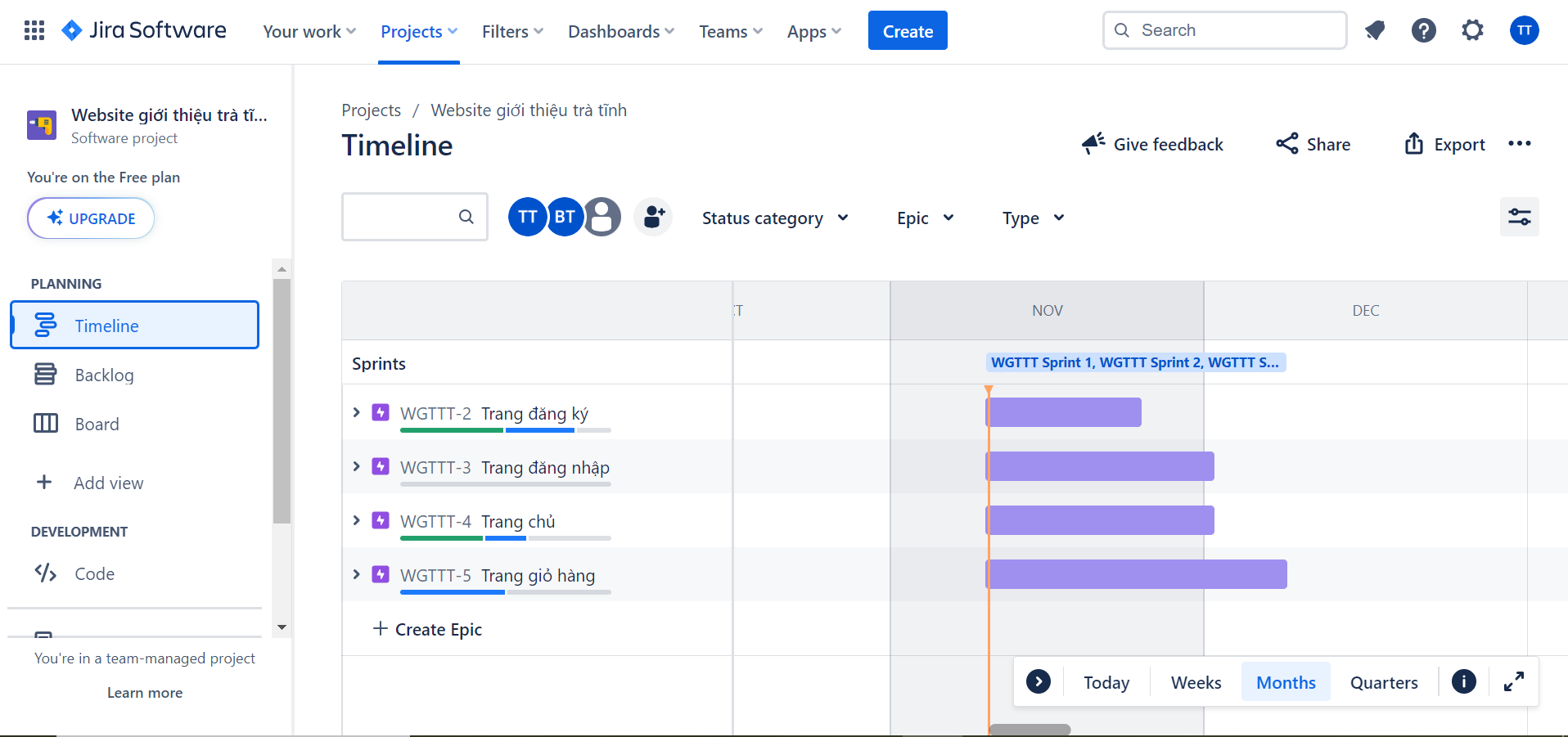








ss



# **CHƯƠNG 4: KẾT LUẬN**

## Kết quả đạt được

* Hiểu được cách thức hoạt động của phần mềm quản lý lỗi Jira
* Nắm bắt được các thông tin cơ bản của phần mềm
* Áp dụng công cụ vào quản lý lỗi của ứng dụng website quảng bá trà.

## Hướng phát triển đề tài

Trong thời gian nghiên cứu và thực hiện đề tài, tác giả đã vạch ra hướng phát triển tiếp theo của đề tài như sau:

* Nghiên cứ cài đặt các module Perl
* Nghiên cứu các chuẩn để đảm bảo chất lượng phần mềm
* Nghiên cứu thêm các phiên bản khác của phần mềm.
* Áp dụng triệt để công cụ Jira trong quản lý và theo dõi lỗi, Đưa việc quản lý lỗi vào những công cụ hỗ trợ như công cụ Jira, Bug Mantis, Testlink, Redmine... Để quản lý một cách dễ dàng hơn.
* Trong tương lai sẽ tiếp tục xây dựng thêm nhưng kịch bản test để phần mềm ít lỗi nhất có thể, thực hiện test những chức năng chưa hoàn thiện, xây dựng trang web hoàn thiện.

TÀI LIỆU THAM KHẢO

1. <https://resources.base.vn/productivity/jira-la-gi-huong-dan-su-dung-jira-cho-nguoi-moi-bat-dau-285>
2. <https://viblo.asia/p/huong-dan-cach-su-dung-jira-don-gian-nhat-cho-developer-6J3ZgOJqZmB?fbclid=IwAR0JH1_s3LdXF6mbcJkNfJM65iYgJ49o6Rsho_X-SdQ8Wwj5NKEf94sAHNA>
3. <https://www.atlassian.com/software/jira/download?fbclid=IwAR0mFgo3NW1gMKjDttWOpKeBceAu0Dx7wv90GRKVj6sguxlNbZzJKrJGMhE>
4. <https://viblo.asia/p/jira-tutorial-gioi-thieu-ve-jira-OeVKBRjdKkW>
5. <http://www.econet.vn/huong-dan-su-dung-jira:-tao-va-lam-viec-voi-sub-task-trong-jira_60.html>
6. <https://topdev.vn/>
7. <https://viblo.asia/p/huong-dan-cach-su-dung-jira-don-gian-nhat-cho-developer-6J3ZgOJqZmB>
8. Giáo trình Đảm bảo chất lượng phần mềm. Khoa Công nghệ Thông tin - UTEHY.
9. Giáo trình Kiểm thử phần mềm. Khoa Công Nghệ Thông Tin – UTEHY.
10. Giáo trình Thiết kế Web. Khoa Công Nghệ Thông Tin - UTEHY

x

x