|  |
| --- |
| TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI  **VIỆN ĐIỆN TỬ - VIỄN THÔNG**  logo_128  ĐỒ ÁN  **TỐT NGHIỆP ĐẠI HỌC**  **Đề tài:**  **THIẾT KẾ HỆ THỐNG QUẢN LÝ THIẾT BỊ Y TẾ**  Sinh viên thực hiện: NGUYỄN THỊ LIỄU  Lớp ĐT 01 – K59  Giảng viên hướng dẫn: ThS. VŨ SONG TÙNG  Hà Nội, 6-2019 |
| TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI  **TRƯỜNG ĐIỆN - ĐIỆN TỬ**  logo_128  **BÁO CÁO ĐỒ ÁN 3**  **Đề tài:**  **THIẾT KẾ HỆ THỐNG QUẢN LÝ THIẾT BỊ Y TẾ**  Sinh viên thực hiện: NGUYỄN THỊ LIỄU  Lớp ĐT 01 – K59  Giảng viên hướng dẫn: ThS. VŨ SONG TÙNG  Cán bộ phản biện:  Hà Nội, 6-2019 |

**ĐÁNH GIÁ QUYỂN ĐỒ ÁN TỐT NGHIỆP**

(Dùng cho giảng viên hướng dẫn)

Tên giảng viên đánh giá: ……ThS. VŨ SONG TÙNG

Họ và tên sinh viên: ………NGUYỄN THỊ LIỄU MSSV: 20142535

Tên đồ án: ……….THIẾT KẾ HỆ THỐNG QUẢN LÝ THIẾT BỊ Y TẾ

**Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:**

Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Có sự kết hợp giữa lý thuyết và thực hành (20)** | | | | | | |
| 1 | Nêu rõ tính cấp thiết và quan trọng của đề tài, các vấn đề và các giả thuyết (bao gồm mục đích và tính phù hợp) cũng như phạm vi ứng dụng của đồ án | 1 | 2 | 3 | 4 | 5 |
| 2 | Cập nhật kết quả nghiên cứu gần đây nhất (trong nước/quốc tế) | 1 | 2 | 3 | 4 | 5 |
| 3 | Nêu rõ và chi tiết phương pháp nghiên cứu/giải quyết vấn đề | 1 | 2 | 3 | 4 | 5 |
| 4 | Có kết quả mô phỏng/thực nghiệm và trình bày rõ ràng kết quả đạt được | 1 | 2 | 3 | 4 | 5 |
| **Có khả năng phân tích và đánh giá kết quả (15)** | | | | | | |
| 5 | Kế hoạch làm việc rõ ràng bao gồm mục tiêu và phương pháp thực hiện dựa trên kết quả nghiên cứu lý thuyết một cách có hệ thống | 1 | 2 | 3 | 4 | 5 |
| 6 | Kết quả được trình bày một cách logic và dễ hiểu, tất cả kết quả đều được phân tích và đánh giá thỏa đáng | 1 | 2 | 3 | 4 | 5 |
| 7 | Trong phần kết luận, tác giả chỉ rõ sự khác biệt (nếu có) giữa kết quả đạt được và mục tiêu ban đầu đề ra đồng thời cung cấp lập luận để đề xuất hướng giải quyết có thể thực hiện trong tương lai | 1 | 2 | 3 | 4 | 5 |
| **Kỹ năng viết quyển đồ án (10)** | | | | | | |
| 8 | Đồ án trình bày đúng mẫu quy định với cấu trúc các chương logic và đẹp mắt (bảng biểu, hình ảnh rõ ràng, có tiêu đề, được đánh số thứ tự và được giải thích hay đề cập đến; căn lề thống nhất, có dấu cách sau dấu chấm, dấu phảy v.v.), có mở đầu chương và kết luận chương, có liệt kê tài liệu tham khảo và có trích dẫn đúng quy định | 1 | 2 | 3 | 4 | 5 |
| 9 | Kỹ năng viết xuất sắc (cấu trúc câu chuẩn, văn phong khoa học, lập luận logic và có cơ sở, từ vựng sử dụng phù hợp v.v.) | 1 | 2 | 3 | 4 | 5 |
| **Thành tựu nghiên cứu khoa học (5)** *(chọn 1 trong 3 trường hợp)* | | | | | | |
| 10a | Có bài báo khoa học được đăng hoặc chấp nhận đăng/Đạt giải SVNCKH giải 3 cấp Viện trở lên/Có giải thưởng khoa học (quốc tế hoặc trong nước) từ giải 3 trở lên/Có đăng ký bằng phát minh, sáng chế | 5 | | | | |
| 10b | Được báo cáo tại hội đồng cấp Viện trong hội nghị SVNCKH nhưng không đạt giải từ giải 3 trở lên/Đạt giải khuyến khích trong các kỳ thi quốc gia và quốc tế khác về chuyên ngành (VD: TI contest) | 2 | | | | |
| 10c | Không có thành tích về nghiên cứu khoa học | 0 | | | | |
| **Điểm tổng** | | **/50** | | | | |
| **Điểm tổng quy đổi về thang 10** | |  | | | | |

***Nhận xét khác*** *(về thái độ và tinh thần làm việc của sinh viên)*

|  |  |
| --- | --- |
|  | Ngày: … / … / 20…  **Người nhận xét**  (Ký và ghi rõ họ tên) |

**ĐÁNH GIÁ QUYỂN ĐỒ ÁN TỐT NGHIỆP**

(Dùng cho cán bộ phản biện)

Giảng viên đánh giá:

Họ và tên sinh viên: …….NGUYỄN THỊ LIỄU MSSV: 20142535

Tên đồ án: …….THIẾT KẾ HỆ THỐNG QUẢN LÝ THIẾT BỊ Y TẾ

**Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:**

Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Có sự kết hợp giữa lý thuyết và thực hành (20)** | | | | | | |
| 1 | Nêu rõ tính cấp thiết và quan trọng của đề tài, các vấn đề và các giả thuyết (bao gồm mục đích và tính phù hợp) cũng như phạm vi ứng dụng của đồ án | 1 | 2 | 3 | 4 | 5 |
| 2 | Cập nhật kết quả nghiên cứu gần đây nhất (trong nước/quốc tế) | 1 | 2 | 3 | 4 | 5 |
| 3 | Nêu rõ và chi tiết phương pháp nghiên cứu/giải quyết vấn đề | 1 | 2 | 3 | 4 | 5 |
| 4 | Có kết quả mô phỏng/thực nghiệm và trình bày rõ ràng kết quả đạt được | 1 | 2 | 3 | 4 | 5 |
| **Có khả năng phân tích và đánh giá kết quả (15)** | | | | | | |
| 5 | Kế hoạch làm việc rõ ràng bao gồm mục tiêu và phương pháp thực hiện dựa trên kết quả nghiên cứu lý thuyết một cách có hệ thống | 1 | 2 | 3 | 4 | 5 |
| 6 | Kết quả được trình bày một cách logic và dễ hiểu, tất cả kết quả đều được phân tích và đánh giá thỏa đáng | 1 | 2 | 3 | 4 | 5 |
| 7 | Trong phần kết luận, tác giả chỉ rõ sự khác biệt (nếu có) giữa kết quả đạt được và mục tiêu ban đầu đề ra đồng thời cung cấp lập luận để đề xuất hướng giải quyết có thể thực hiện trong tương lai | 1 | 2 | 3 | 4 | 5 |
| **Kỹ năng viết quyển đồ án (10)** | | | | | | |
| 8 | Đồ án trình bày đúng mẫu quy định với cấu trúc các chương logic và đẹp mắt (bảng biểu, hình ảnh rõ ràng, có tiêu đề, được đánh số thứ tự và được giải thích hay đề cập đến; căn lề thống nhất, có dấu cách sau dấu chấm, dấu phảy v.v.), có mở đầu chương và kết luận chương, có liệt kê tài liệu tham khảo và có trích dẫn đúng quy định | 1 | 2 | 3 | 4 | 5 |
| 9 | Kỹ năng viết xuất sắc (cấu trúc câu chuẩn, văn phong khoa học, lập luận logic và có cơ sở, từ vựng sử dụng phù hợp v.v.) | 1 | 2 | 3 | 4 | 5 |
| **Thành tựu nghiên cứu khoa học (5)** *(chọn 1 trong 3 trường hợp)* | | | | | | |
| 10a | Có bài báo khoa học được đăng hoặc chấp nhận đăng/Đạt giải SVNCKH giải 3 cấp Viện trở lên/Có giải thưởng khoa học (quốc tế hoặc trong nước) từ giải 3 trở lên/Có đăng ký bằng phát minh, sáng chế | 5 | | | | |
| 10b | Được báo cáo tại hội đồng cấp Viện trong hội nghị SVNCKH nhưng không đạt giải từ giải 3 trở lên/Đạt giải khuyến khích trong các kỳ thi quốc gia và quốc tế khác về chuyên ngành (VD: TI contest) | 2 | | | | |
| 10c | Không có thành tích về nghiên cứu khoa học | 0 | | | | |
| **Điểm tổng** | | **/50** | | | | |
| **Điểm tổng quy đổi về thang 10** | |  | | | | |

***Nhận xét khác của cán bộ phản biện***

|  |  |
| --- | --- |
|  | Ngày: … / … / 20…  **Người nhận xét**  (Ký và ghi rõ họ tên) |

**LỜI NÓI ĐẦU**

Công tác quản lý thông tin thiết bị ở bệnh viện hiện nay thường được tiến hành trên sổ sách, nội dung quản lý thông tin thường thiếu khoa học. Cách quản lý này không đáp ứng được nhu cầu của các cán bộ y tế, đặc biệt là khi trang thiết bị y tế đang đóng vai trò quan trọng trong việc duy trì sự ổn định của bệnh viện. Chính vì lẽ đó, em đã lựa chọn đề tài:“ **Thiết kế hệ thống quản lý thiết bị y tế** ”, giúp cho quá trình quản lý thiết bị y tế được dễ dàng, chính xác hơn, hỗ trợ tốt cho bệnh viện, trung tâm y tế hay các phòng khám.

Trong quá trình thực hiện đồ án, em xin gửi lời cảm ơn chân thành tới ThS. Vũ Song Tùng, Viện Điện tử - Viễn thông, Trường Đại học Bách Khoa Hà Nội, đã hướng dẫn rất tận tình, nhiệt tình chỉ bảo và đưa ra hướng nghiên cứu tối ưu nhất để em có thể thực hiện được các yêu cầu của đề tài, từ đó có thể hoàn thiện đồ án một cách tốt nhất.

Trong quá trình thực hiện đồ án không tránh khỏi những sai sót và hạn chế nhất định. Vì vậy, em rất mong nhận được sự góp ý và bổ sung của các thầy cô để đề tài được tối ưu và hoàn thiện nhất có thể.

Em xin chân thành cảm ơn!

**LỜI CAM ĐOAN**

Tôi là Nguyễn Thị Liễu, mã số sinh viên 20142535, sinh viên lớp Điện tử 01, khóa 59. Người hướng dẫn là Th.S Vũ Song Tùng. Tôi xin cam đoan toàn bộ nội dung được trình bày trong đồ án *Thiết kế hệ thống quản lý thiết bị y tế* là kết quả quá trình tìm hiểu và nghiên cứu của tôi. Các dữ liệu được nêu trong đồ án là hoàn toàn trung thực, phản ánh đúng kết quả thực tế. Mọi thông tin trích dẫn đều tuân thủ các quy định về sở hữu trí tuệ, các tài liệu tham khảo được liệt kê rõ ràng. Tôi xin chịu hoàn toàn trách nhiệm với những nội dung được viết trong đồ án này.

Hà Nội, ngày 06 tháng 06 năm 2019

**Người cam đoan**

**Nguyễn Thị Liễu**

**MỤC LỤC**

[DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT iii](#_Toc11065133)

[DANH MỤC HÌNH VẼ iv](#_Toc11065134)

[DANH MỤC BẢNG BIỂU vi](#_Toc11065135)

[TÓM TẮT ĐỒ ÁN vii](#_Toc11065136)

[CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI 1](#_Toc11065137)

[1.1 Tổng quan hệ thống 1](#_Toc11065138)

[1.1.1 Đặt vấn đề 1](#_Toc11065139)

[1.1.2 Mục đích nghiên cứu 1](#_Toc11065140)

[1.1.3 Phương pháp nghiên cứu 2](#_Toc11065141)

[1.1.4 Kết luận 2](#_Toc11065142)

[1.2 Hệ quản trị cơ sở dữ liệu phân tán 2](#_Toc11065143)

[1.2.1 Khái niệm hệ quản trị CSDLPT 2](#_Toc11065144)

[1.2.2 Sự cần thiết của hệ CSDLPT 4](#_Toc11065145)

[1.2.3 Đặc điểm của CSDLPT 5](#_Toc11065146)

[1.2.4 So sánh CSDLPT và Cơ sở dữ liệu tập trung 8](#_Toc11065147)

[1.3 Web API 9](#_Toc11065148)

[1.3.1 Khái niệm API 9](#_Toc11065149)

[1.3.2 Khái niệm Web API 11](#_Toc11065150)

[1.3.3 Ưu điểm của Web API 11](#_Toc11065151)

[1.4 Kết luận 12](#_Toc11065152)

[CHƯƠNG 2. THIẾT KẾ HỆ THỐNG QUẢN LÝ THIẾT BỊ Y TẾ 13](#_Toc11065153)

[2.1 Phân tích hệ thống 13](#_Toc11065154)

[2.1.1 Tổng quan hệ thống 13](#_Toc11065155)

[2.1.2 Các đối tượng người dùng 13](#_Toc11065156)

[2.1.3 Các yêu cầu chức năng 14](#_Toc11065157)

[2.1.4 Các yêu cầu phi chức năng 16](#_Toc11065158)

[2.2 Thiết kế phần Window App 17](#_Toc11065159)

[2.2.1 Các UseCase trong Window App 17](#_Toc11065160)

[2.2.2 Mô tả các hoạt động trên Window App 20](#_Toc11065161)

[2.3 Thiết kế phần Mobile App 23](#_Toc11065162)

[2.3.1 Các UseCase trong Mobile App 23](#_Toc11065163)

[2.3.2 Mô tả các hoạt động trên Mobile App 25](#_Toc11065164)

[2.4 Cơ sở dữ liệu MongoDB 28](#_Toc11065165)

[2.4.1 Khái niệm MongoDB và NoSQL 28](#_Toc11065166)

[2.4.2 Đặc điểm của MongoDB 29](#_Toc11065167)

[2.4.3 Kiến trúc MongoDB 30](#_Toc11065168)

[2.4.4 Ưu điểm của việc sử dụng cơ sở dữ liệu MongoDB 30](#_Toc11065169)

[2.5 Kết luận 31](#_Toc11065170)

[CHƯƠNG 3. XÂY DỰNG BACKEND APPLICATION 32](#_Toc11065171)

[3.1 Xây dựng bộ API trên Server 32](#_Toc11065172)

[3.1.1 API Acount 32](#_Toc11065173)

[3.1.2 API Thiết bị 33](#_Toc11065174)

[3.1.3 API Bảo trì 33](#_Toc11065175)

[3.1.4 API Danh mục 34](#_Toc11065176)

[3.1.5 API Báo hỏng 35](#_Toc11065177)

[3.1.6 API Khách hàng 36](#_Toc11065178)

[3.1.7 API Database 36](#_Toc11065179)

[3.1.8 API Đơn vị 36](#_Toc11065180)

[3.1.9 API User 37](#_Toc11065181)

[3.1.10 API Xử lý 37](#_Toc11065182)

[3.1.11 API Báo cáo 37](#_Toc11065183)

[3.2 Kiểm thử 38](#_Toc11065184)

[3.2.1 Giao diện trên Web 38](#_Toc11065185)

[3.2.2 Giao diện trên Window 39](#_Toc11065186)

[3.2.3 Giao diện trên Mobile App 42](#_Toc11065187)

[3.3 Đánh giá kết quả 44](#_Toc11065188)

[3.4 Kết luận 44](#_Toc11065189)

[KẾT LUẬN 45](#_Toc11065190)

[Kết luận chung 45](#_Toc11065191)

[TÀI LIỆU THAM KHẢO 46](#_Toc11065192)

# DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **STT** | **Kí hiệu** | **Thuật ngữ đầy đủ** |
| 1 | TBYT | Thiết bị y tế |
| 2 | CSDL | Cơ sở dữ liệu |
| 3 | CSDLPT | Cơ sở dữ liệu phân tán |
| 4 | JSON | JavaScript Object Notation |
| 5 | BSON | Binary JSON |
| 6 | API | Application Programing Interface |
| 7 | TPVT | Trưởng phòng vật tư |
| 8 | KTV | Kỹ thuật viên |
| 9 | TBYT | Thiết bị y tế |
| 10 | CPU | Central Processing Unit |
| 11 | XML | Extensible Markup Language |
| 12 | SQL | Structured Query Language |
| 13 | CSDLTT | Cơ sở dữ liệu tập trung |
| 14 | NOSQL | Not Only SQL |
| 15 | MONGODB | Mongo Document Database |

# DANH MỤC HÌNH VẼ

[Hình 1. 1 Mô hình tổng quan hệ thống 2](#_Toc11064466)

[Hình 1. 2 Hệ quản trị cơ sở dữ liệu phân tán 3](file:///C:\Users\lieu%20thi\Desktop\ĐATN_1.docx#_Toc11064467)

[Hình 1. 3 Tổng quan về API 10](#_Toc11064468)

[Hình 1. 4 Mô hình WebAPI 11](#_Toc11064469)

[Hình 2. 1 Sơ đồ tổng quan hệ thống 13](#_Toc11064376)

[Hình 2. 2 UseCase TPVT trên Window 18](#_Toc11064377)

[Hình 2. 3 UseCase Bác sĩ trên Window 19](#_Toc11064378)

[Hình 2. 4 UseCase KTV trên Window 20](#_Toc11064379)

[Hình 2. 5 Activity Báo hỏng trên Window 21](#_Toc11064380)

[Hình 2. 6 Activity Xác nhận sử dụng trên Window 22](#_Toc11064381)

[Hình 2. 7 Activity Nhận sửa trên Window 23](#_Toc11064382)

[Hình 2. 8 UseCase Bác sĩ trên Mobile App 24](#_Toc11064383)

[Hình 2. 9 UseCase TPVT và KTV trên Mobile App 25](#_Toc11064384)

[Hình 2. 10 Activity Báo hỏng trên Mobile App 26](#_Toc11064385)

[Hình 2. 11 Activity Xác nhận sử dụng trên Mobile App 27](#_Toc11064386)

[Hình 2. 12 Activity Nhận sửa trên Mobile 28](#_Toc11064387)

[Hình 3. 1 Giao diện Web quản lý thiết bị 38](#_Toc11064388)

[Hình 3. 2 Giao diện quản lý thiết bị trên Window 39](#_Toc11064389)

[Hình 3. 3 Giao diện lịch bảo trì trên Window 40](#_Toc11064390)

[Hình 3. 4 Giao diện thông tin thiết bị cần bảo trì trên Window App 40](#_Toc11064391)

[Hình 3. 5 Giao diện danh sách các thiết bị chờ bảo dưỡng trên Window App 41](#_Toc11064392)

[Hình 3. 6 Giao diện màn hình cập nhật thiết bị trên Window App 41](#_Toc11064393)

[Hình 3. 7 Giao diện phần thống kê thiết bị từng đơn vị trên Window App 42](#_Toc11064394)

[Hình 3. 8 Giao diện màn hình đăng nhập trên Mobile App 42](#_Toc11064395)

[Hình 3. 9 Danh sách các thiết bị hỏng trên Mobile App 43](#_Toc11064396)

[Hình 3. 10 Màn hình Mobile App sau khi quét mã QR code 43](#_Toc11064397)

[Hình 3. 11 Giao diện đăng xuất và thay đổi mật khẩu trên Mobile App 44](#_Toc11064398)

# DANH MỤC BẢNG BIỂU

[Bảng 2. 1 So sánh SQL Database và MongoDB 29](#_Toc11077178)

[Bảng 3. 1 API Account 32](#_Toc11064545)

[Bảng 3. 2 API Thiết bị 33](#_Toc11064546)

[Bảng 3. 3 API Bảo trì 34](#_Toc11064547)

[Bảng 3. 4 API Danh mục 35](#_Toc11064548)

[Bảng 3. 5 API Báo hỏng 35](#_Toc11064549)

[Bảng 3. 6 API Khách hàng 36](#_Toc11064550)

[Bảng 3. 7 API Database 36](#_Toc11064551)

[Bảng 3. 8 API Đơn vị 36](#_Toc11064552)

[Bảng 3. 9 API User 37](#_Toc11064553)

[Bảng 3. 10 API Xử lý 37](#_Toc11064554)

[Bảng 3. 11 API Báo cáo 38](#_Toc11064555)

# 

# TÓM TẮT ĐỒ ÁN

Nội dung chính của đề tài là trình bày các vấn đề liên quan đến cơ sở dữ liệu MongoDB và các bước phân tích, thiết kế hệ thống quản lý thiết bị y tế trên ba phần Server, Window App và Mobile App. Cụ thể:

**Chương 1:** Trình bày về tổng quan đề tài, về yêu cầu, mục đích và ứng dụng của đề tài vào thực tiễn, đồng thời trình bày cơ sơ lý thuyết phục vụ quá trình thiết kế và xây dựng hệ thống.

**Chương 2:** Trình bày các bước phân tích hệ thống, xây dựng cơ sở dữ liệu của hệ thống quản lý thiết bị y tế.

**Chương 3:** Trình bày việc xây dựng bộ API trên Server và kiểm thử các kết quả trên Window App và Mobile App.

**Phần kết luận:** Trình bày những kết quả đã thực hiện được và chưa thực hiện được, từ đó đưa ra hướng phát triển thêm cho đề tài.

**ABSTRACT**

The main content of the topic is to present issues related to MongoDB database and analysis steps, design medical device management system on three parts Server, Window App and Mobile App. Specifically:

**Chapter 1:** Presentation of the topic overview, the requirements, purposes and applications of the topic into practice, and present the theoretical basis for the process of designing and building the system.

**Chapter 2:** Presentation of steps to analyze the system, build a database of medical equipment management system.

**Chapter 3:** Presentation of building APIs on Server and testing results on Window App and Mobile App.

**Concluding remarks:** Presenting the results that have been implemented and have not been implemented, thereby giving further development direction for the topic.

# TỔNG QUAN ĐỀ TÀI

Chương 1 đề cập đến tổng quan đề tài quản lý thiết bị y tế và các cơ sở lý thuyết về hệ cơ sở dữ liệu phân tán, so sánh cơ sở dữ liệu tập trung với cơ sở dữ liệu phân tán để có thể thấy rõ lợi ích khi sử dụng cơ sở dữ liệu phân tán trong hệ thống quản lý thiết bị y tế.

## Tổng quan hệ thống

### Đặt vấn đề

Việc quản lý các thiết bị y tế tại các bệnh viện lớn nhỏ hay tại các phòng khám luôn là điều vô cùng quan trọng, giúp cho các cán bộ y tế có thể kiểm soát được số lượng cũng như chất lượng của thiết bị, từ đó có thể đảm bảo hiệu quả khám chữa bệnh được chính xác. Tuy nhiên việc quản lý các thiết bị y tế chưa bao giờ là dễ dàng, bởi đó là công việc đòi hỏi nhiều công sức, sự tỉ mỉ, cẩn thận, nhưng đôi khi lại không đem lai hiệu quả như mong muốn. Chính vì vậy em đã chọn đề tài: “ ***Thiết kế hệ thống quản lý thiết bị y tế***” với mong muốn tiết kiệm công sức, linh hoạt thời gian cho các cán bộ y tế trong việc quản lý và vận hành các thiết bị, đồng thời nâng cao hiệu quả khám chữa bệnh tại các bệnh viện, trung tâm y tế hay các phòng khám.

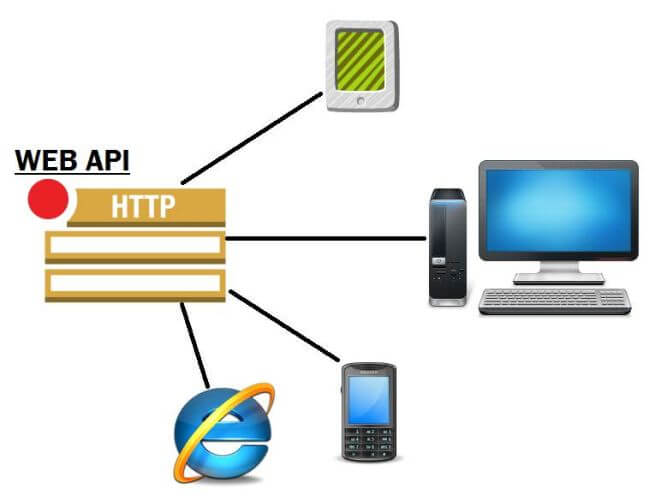
### Mục đích nghiên cứu

Đề tài được nghiên cứu và thực hiện với mục đích xây dựng và phát triển một hệ thống quản lý thiết bị y tế trên nền tảng window app, web và mobile app giúp tiết kiệm thời gian, công sức trong việc quản lý các thiết bị y tế, từ đó nâng cao hiệu quả làm việc cho các cán bộ y tế và nâng cao chất lượng khám chữa bệnh trong các bệnh viện, trung tâm y tế hay phòng khám.

Mô hình tổng quan hệ thống được thể hiện như hình 1.1 [2].

Sau khi thực hiện, đề tài đặt ra những chức năng sau:

* Tìm hiểu và xây dựng một hệ thống toàn diện giúp người dùng, cụ thể là các cán bộ y tế có thể sử dụng dễ dàng trên cả window và mobile app, đem lại hiệu quả nhất định trong việc quản lý và vận hành trang thiết bị y tế trong bệnh viện.
* Xây dựng được một cơ sở dữ liệu an toàn, bảo mật.



Hình 1. 1 Mô hình tổng quan hệ thống

### Phương pháp nghiên cứu

Trong đề tài sử dụng các phương pháp nghiên cứu:

* Phương pháp phân tích và tổng hợp lý thuyết: nghiên cứu các tài liệu về cơ sở dữ liệu phân tán, cơ sở dữ liệu MongoDB để nắm vững kiến thức từ đó xây dựng được bộ cơ sở dữ liệu cho việc quản lý thiết bị y tế.
* Phương pháp thực nghiệm: Xem xét một số cơ sở dữ liệu đã được áp dụng trước đó để rút ra kinh nghiệm cũng như những yêu cầu đề ra cho hệ thống quản lý thiết bị y tế.

### Kết luận

Phần mở đầu giới thiệu tổng quan về đề tài *Thiết kế hệ thống quản lý thiết bị y tế*. Việc tìm hiểu và triển khai thiết kế hệ thống quản lý thiết bị y tế giúp các cán bộ y tế dễ dàng quản lý thiết bị, tiết kiệm thời gian, công sức và nâng cao hiệu quả vận hành.

## Hệ quản trị cơ sở dữ liệu phân tán

### Khái niệm hệ quản trị CSDLPT

Một cách trực quan, một CSDL phân tán là một bộ sưu tập các loại dữ liệu có liên kết logic với nhau và được phân bố vật lý trên nhiều máy chủ của mạng máy tính. Khái niệm hệ CSDLPT bao gồm cả khái niệm CSDL và hệ quản trị CSDLPT [1].

Định nghĩa này nhấn mạnh hai khía cạnh quan trọng của CSDLPT:

* Tính phân tán
* Sự tương quan logic
  + - * Tính phân tán: thực tế dữ liệu không cư trú trên cùng một site, vì vậy có thể phân biệt một CSDLPT với cơ sở dữ liệu tập trung (CSDLTT).



**Network**

Database

Database

Hệ quản trị CSDL phân tán

Trình quản lý truyền thông

Trình quản lý dữ liệu phân tán

Trình quản lý các ứng dụng

Hệ quản trị CSDL phân tán

Trình quản lý truyền thông

Trình quản lý dữ liệu phân tán

Trình quản lý các ứng dụng

Hình 1. 2 Hệ quản trị cơ sở dữ liệu phân tán

* + - * Sự tương quan logic: các loại dữ liệu có một số tính chất ràng buộc lẫn nhau, như vậy có thể phân biệt CSDLPT với tập các CSDL địa phương hoặc với các tệp lưu trữ trên các site khác nhau.

Hệ quản trị CSDL phân tán là hệ thống phần mềm cho phép quản trị CSDL phân tán và làm cho sự phân tán đó là trong suốt đối với người sử dụng. Nói cách khác CSDL phân tán là CSDL được phân tán một cách vật lý nhưng được thống nhất tổ chức như là một CSDL duy nhất [1].

Như vậy sự phân tán dữ liệu là trong suốt đối với người sử dụng. Việc quản lý các dữ liệu phân tán đòi hỏi mỗi trạm (site) cài đặt các thành phần hệ thống sau:

* Thành phần quản trị CSDL (Database Management DM).
* Thành phần truyền dữ liệu (Data Communication DC).
* Từ điển dữ liệu (Data Dictionary DD): thông tin về sự phân tán dữ liệu trên mạng
* Thành phần CSDLPT (Distributed Database DDB)

Các dịch vụ của hệ thống trên bao gồm:

* Các ứng dụng truy nhập CSDL từ xa .
* Cung cấp các mức trong suốt phân tán.
* Hỗ trợ quản trị và điều khiển CSDL, bao gồm các bộ công cụ, thu thập thông tin từ các trình tiện ích, cung cấp cách nhìn tổng quan về các file dữ liệu trên mạng.
* Khả năng mở rộng với các hệ thống khác nhau.
* Cung cấp khả năng điều khiển đồng thời và phục hồi các giao tác phân tán.

Trong mô hình cơ sở dữ liệu phân tán, bản thân cơ sở dữ liệu có ở trên nhiều máy tính khác nhau. Như vậy, đặc trưng của cơ sở dữ liệu phân tán là các CSDL được phân bố trên mạng máy tính và có quan hệ với nhau về mặt logic.

Hệ CSDL phân tán không đơn thuần bao gồm nhiều file dữ liệu được tổ chức lưu trữ riêng lẻ trên các thiết bị nhớ của mạng máy tính.

Để tạo một hệ CSDL phân tán, các file không chỉ có quan hệ với nhau về mặt logic mà còn cần có một cấu trúc giao diện chung giữa chúng để file có thể truy nhập lẫn nhau.

### Sự cần thiết của hệ CSDLPT

Công nghệ cơ sở dữ liệu phân tán đã trở thành một lĩnh vực quan trọng trong công nghệ thông tin, tính cần thiết của nó ngày càng được nâng cao. Có nhiều nguyên nhân thúc đẩy sự phát triển của các hệ CSDL:

##### Sự phát triển của các cơ cấu tổ chức

Cùng với sự phát triển của xã hội, nhiều cơ quan, xí nghiệp có cơ cấu tổ chức không tập trung, hoạt động phân tán trên phạm vi rộng. Vì vậy thiết kế và cài đặt cơ sở dữ liệu phân tán là phù hợp, đáp ứng mọi nhu cầu truy xuất và khai thác dữ liệu. Cơ cấu tổ chức và vấn đề kinh tế là một trong những nguyên nhân quan trọng nhất của sự phát triển cơ sở dữ liệu phân tán.

##### Giảm chi phí truyền thông

Việc sử dụng một số ứng dụng mang tính địa phương sẽ làm giảm chi phí truyền thông.

Bởi vậy, việc tối ưu hóa tính địa phương của các ứng dụng là một trong những mục tiêu chính của việc thiết kế và cài đặt một CSDLPT.

##### Độ tin cậy và tính sẵn sàng

Cách tiếp cận CSDLPT, cho phép truy nhập độ tin cậy và tính sẵn sàng cao hơn. Tuy nhiên, những lỗi xuất hiện trong một CSDLPT có thể xảy ra nhiều hơn vì số các thành phần cấu thành lớn hơn, nhưng ảnh hưởng của lỗi ít hơn.

### Đặc điểm của CSDLPT

Cơ sở dữ liệu phân tán không đơn giản là sự phân bố của các cơ sở dữ liệu, bởi vì cơ sở dữ liệu phân tán có nhiều đặc điểm khác biệt so với cơ sở dữ liệu tập trung truyền thống.

*1.2.3.1 Điều khiển tập trung*

Điều khiển tập trung (Centralized Control) là một đặc điểm của cơ sở dữ liệu tập trung, toàn bộ dữ liệu được tập trung lại nhằm tránh sự dư thừa dữ liệu, đảm bảo được tính độc lập của dữ liệu. Dữ liệu được quản lý tập trung bởi người quản trị cơ sở dữ liệu nhằm bảo đảm sự an toàn của dữ liệu. Trong các cơ sở dữ liệu phân tán , sự điều khiển được thực hiện theo một cấu trúc điều khiển phân cấp bao gồm hai loại người quản trị cơ sở dữ liệu:

* Người quản trị cơ sở dữ liệu toàn cục (Global Database Administrator) là người có trách nhiệm chính về toàn bộ cơ sở dữ liệu phân tán..
* Người quản trị cơ sở dữ liệu cục bộ (Local Database Administrator) là người có trách nhiệm về cơ sở dữ liệu cục bộ của họ.

Tuy nhiên, những người quản trị cơ sở dữ liệu cục bộ cần phải có những quyền độc lập riêng về cơ sở dữ liệu cục bộ của mình mà người quản trị cơ sở dữ liệu toàn cục hoàn toàn không có những quyền này và sự phối hợp giữa các vị trí được thực hiện bởi chính những người quản trị cục bộ. Đặc điểm này được gọi là sự độc lập vị trí.

Các cơ sở dữ liệu phân tán có thể khác nhau rất nhiều về mức độ độc lập vị trí. Từ sự độc lập vị trí hoàn toàn (không có người quản trị cơ sở dữ liệu tập trung) đến sự điều khiển tập trung hoàn toàn.

*1.2.3.2 Độc lập dữ liệu*

Độc lập dữ liệu (Data Independence) là một đặc điểm của cơ sở dữ liệu. Độc lập dữ liệu có nghĩa là tổ chức lưu trữ dữ liệu là trong suốt đối với người lập trình ứng dụng.

Ưu điểm của độc lập dữ liệu là các chương trình không bị ảnh hưởng bởi những thay đổi về tổ chức lưu trữ vật lý của dữ liệu.

Trong các hệ cơ sở dữ liệu phân tán, độc lập dữ liệu cũng quan trọng như trong các cơ sở dữ liệu tập trung. Tuy nhiên, một đặc điểm mới được đưa vào trong khái niệm thông thường của độc lập dữ liệu là sự trong suốt phân tán (Distribution Transparency). Nhờ sự trong suốt phân tán mà các chương trình ứng dụng có thể được viết giống như trong cơ sở dữ liệu không được phân tán.

Vì vậy, tính đúng đắn của các chương trình ứng dụng không bị ảnh hưởng bởi sự di chuyển dữ liệu từ một vị trí này đến một vị trí khác.

Độc lập dữ liệu trong cơ sở dữ liệu tập trung được thể hiện thông qua một kiến trúc nhiều mức, các mức này có những mô tả khác nhau về dữ liệu và những ánh xạ biến đổi giữa các mức.

*1.2.3.3 Giảm dư thừa dữ liệu*

Trong các cơ sở dữ liệu tập trung, sự dư thừa dữ liệu được giảm thiểu, vì tránh sự không nhất quán giữa nhiều bản sao bằng cách chỉ có một bản sao và tiết kiệm vùng nhớ lưu trữ. Các ứng dụng chia sẻ chung, truy xuất đến các tập tin dữ liệu.

Tuy nhiên, trong các cơ sở dữ liệu phân tán, sự dư thừa dữ liệu là một đặc điểm cần thiết, vì các lý do sau:

Làm tăng tính cục bộ của các ứng dụng nếu dữ liệu được nhân bản tại tất cả các vị trí mà ứng dụng cần dữ liệu này. Khi đó, các ứng dụng cục bộ được thực hiện nhanh hơn vì không cần phải truy xuất dữ liệu từ xa.

Làm tăng tính sẵn sàng của hệ thống ứng dụng, vì một vị trí có sự cố sẽ không làm ngưng sự thực hiện của các ứng dụng ở những vị trí khác nếu dữ liệu tại vị trí bị hỏng được nhân bản tại các vị trí khác.

Tuy nhiên, sự nhân bản dữ liệu cần phải xem xét kỹ lưỡng dựa vào hai loại ứng dụng cơ bản, đó là ứng dụng chỉ đọc và ứng dụng cập nhật.

Sự nhân bản dữ liệu giúp cho các ứng dụng chỉ đọc được thực hiện nhanh hơn, nhưng nó làm cho các ứng dụng cập bị thực hiện lâu hơn vì phải cập nhật dữ liệu tại các vị trí được nhân bản.

Như vậy, sự nhân bản dữ liệu sẽ là một ưu điểm nếu hệ thống có rất nhiều ứng dụng chỉ đọc và có rất ít ứng dụng cập nhật. Trong trường hợp ngược lại thì sự nhân bản dữ liệu lại là một nhược điểm.

*1.2.3.4 Độ tin cậy qua các giao dịch phân tán*

Hệ quản trị CSDL phân tán cải thiện độ tin cậy qua các giao dịch phân tán, vì các thành phần được nhân bản hạn chế được các vị trí lỗi riêng lẻ. Lỗi của trạm riêng, hoặc lỗi của truyền thông làm cho một hoặc nhiều trạm mất liên lạc, không đủ để phá vỡ toàn bộ hệ thống. Trong trường hợp CSDL phân tán, điều này nghĩa là một số dữ liệu không thể truy nhập được, nhưng nếu biết cách hỗ trợ cho các giao dịch phân tán và các giao thức ứng dụng, thì người sử dụng vẫn có thể truy nhập được tới phần khác trong CSDL phân tán.

Giao dịch là một đơn vị tính toán cơ bản, nhất quán và tin cậy, bao gồm một chuỗi các thao tác CSDL được thực hiện chuyển từ trạng thái CSDL nhất quán này sang trạng thái CSDL nhất quán khác ngay cả khi có một số giao dịch được thực hiện đồng thời và thậm chí cả khi xảy ra lỗi. Vì vậy, hệ quản trị CSDL phải hỗ trợ đầy đủ cho giao dịch đảm bảo rằng việc thực thi đồng thời các giao dịch của người sử dụng sẽ không vi phạm tính nhất quán của CSDL trong khi hệ thống có lỗi, với điều kiện là giao dịch được thực hiện chính xác, nghĩa là tuân theo các qui tắc toàn vẹn của CSDL.

*1.2.3.5 Cải tiến hiệu năng*

Hiệu năng của CSDL phân tán được cải tiến dựa vào hai điểm:

* Hệ quản trị CSDL phân tán có khả năng phân mảnh CSDL khái niệm và cho phép cục bộ hoá dữ liệu.

Có hai ưu điểm nổi bật:

* Vì mỗi trạm chỉ xử lý một phần CSDL, sự tranh chấp về CPU và các dịch vụ vào/ra không nghiêm trọng như trong các hệ CSDL tập trung.
* Tính cục bộ làm giảm trễ truy nhập từ xa thường gặp trên các mạng diện rộng.

Hầu hết các hệ CSDL phân tán được cấu trúc nhằm tận dụng tối đa những ưu điểm của tính cục bộ dữ liệu.

Lợi ích đầy đủ của việc giảm tranh chấp và giảm chi phí truyền chỉ có thể có được bằng cách phân mảnh và phân tán dữ liệu hợp lý.

* Tính song song của các hệ thống phân tán có thể được khai thác để thực hiện song song liên truy vấn và truy vấn nội bộ. Liên truy vấn song song là khả năng thực hiện nhiều truy vấn tại cùng thời điểm, còn nội truy vấn song song là phương pháp tách một truy vấn đơn thành các truy vấn con và mỗi truy vấn con được thực hiện tại các trạm khác nhau, truy nhập các phần khác nhau của CSDL phân tán.

*1.2.3.6 Dễ dàng mở rộng hệ thống*

Trong môi trường phân tán, dễ dàng tăng kích thước dữ liệu. và hiếm khi cần sửa đổi trong các hệ thống lớn. Việc mở rộng thường có thể được thực hiện bằng cách tăng khả năng lưu trữ và xử lý của mạng. Rõ ràng là không thể có được sự gia tăng “khả năng” một cách tuyến tính, vì điều này phụ thuộc vào chi phí phân tán. Tuy nhiên, vẫn có thể có những cải tiến có ý nghĩa. Khả năng mở rộng hệ thống dễ dàng mang tính kinh tế, chi phí giảm.

### So sánh CSDLPT và Cơ sở dữ liệu tập trung

CSDL phân tán không đơn giản là những sự thực hiện phân tán của CSDL tập trung, bởi vì chúng cho phép thiết kế các đặc trưng khác với CSDL tập trung truyền thống. Các đặc điểm tiêu biểu của CSDL truyền thống gồm: Điều khiển tập trung, độc lập dữ liệu, giảm dư thừa dữ liệu, biệt lập và bảo mật dữ liệu.

*1.2.4.1 Điều khiển tập trung*

Trong CSDL tập trung: Khả năng điều khiển tập trung trên toàn nguồn tài nguyên thông tin của tổ chức, được xem là động cơ mạnh nhất cho việc ra đời CSDL. Chúng được phát triển như là sự tiến hoá của hệ thống thông tin mà trong đó mỗi ứng dụng có các tập tin riêng của nó.Trong CSDL phân tán: Ý niệm về điều khiển tập trung ít được nhấn mạnh hơn, điều này phụ thuộc vào kiến trúc của CSDL phân tán.

*1.2.4.2 Độc lập dữ liệu*

Trong CSDL phân tán, độc lập dữ liệu cũng quan trọng giống như trong CSDL truyền thống. Tuy nhiên, một khía cạnh mới được thêm vào trong ý niệm của độc lập dữ liệu là trong suốt phân tán. Với trong suốt phân tán chúng ta hiểu rằng các chương trình ứng dụng có thể sử dụng CSDL như là nó không được tổ chức phân tán. Vì thế sự chính xác của chương trình không bị ảnh hưởng bởi việc dịch chuyển dữ liệu từ trạm này đến trạm khác, tuy nhiên tốc độ thực hiện của chúng bị ảnh hưởng.

*1.2.4.3 Giảm dư thừa dữ liệu*

Trong CSDL truyền thống, dữ liệu dư thừa được giảm đến mức tối thiểu bởi hai lý do: Sự không tương thích giữa nhiều bản sao của cùng một tập dữ liệu. Tiết kiệm không gian lưu trữ bằng cách loại bỏ các dư thừa. Việc giảm dư thừa dữ liệu có thể đạt được bằng cách chia sẻ dữ liệu cho phép nhiều ứng dụng truy cập cùng các bản tin và bản ghi.Trong CSDL phân tán, việc giảm dư thừa phức tạp hơn vì ngoài hai lý do trên còn nhiều lý do để giảm dư thừa như:Hoạt động của các trình ứng dụng có thể bị tăng lên khi dữ liệu được sao lại tất cả các vị trí nơi trình ứng dụng cần nó. Tính thường trực của hệ thống sẽ tăng lên bởi vì khi có lỗi xảy ra ở một trạm nào đó sẽ không dừng việc thực hiện các ứng dụng của trạm khác nếu dữ liệu đã được sao chép lại.

*1.2.4.4 Bảo mật*

Trong CSDL truyền thống, hệ quản trị cơ sở dữ liệu tập trung có thể bảo đảm chỉ truy cập đến dữ liệu đã được uỷ quyền. Trong CSDL phân tán, hệ quản trị dữ liệu địa phương thực chất phải đương đầu với các vấn đề giống như hệ quản trị cơ sở dữ liệu trong CSDL truyền thống. Tuy nhiên, hai khía cạnh đặc biệt sau đây của CSDL phân tán cần phải được xem xét: Trong CSDL phân tán với một mức độ tự trị rất cao của các địa phương, người chủ dữ liệu địa phương cảm giác được bảo vệ tốt hơn vì họ có thể tự chủ thực hiện bảo vệ thay vì phụ thuộc vào người quản trị CSDL trung tâm. Vấn đề bảo mật là bản chất trong hệ phân tán nói chung, vì các mạng truyền thông diện rộng cho phép nhiều người cập nhật và khai thác dữ liệu nên cần được bảo vệ.

## Web API

### Khái niệm API

**API**là viết tắt của **Application Programming Interface**(giao diện lập trình ứng dụng), là phần mềm trung gian cho phép 2 ứng dụng giao tiếp với nhau [4].

API là lớp chuyên xử lý các thao tác người dùng, nhận request từ người dùng, xử lý và trả về dữ liệu Database, sau đó lại lấy dữ liệu từ Database gửi ngược lại người dùng ở tầng giao diện.

Windows có nhiều API, và Twitter cũng có web API, tuy nhiên chúng thực hiện các chức năng khác nhau, với mục tiêu khác nhau.

API chính là một phần mềm giao tiếp được sử dụng bởi các ứng dụng khác nhau.



Hình 1. 3 Tổng quan về API

Mỗi bộ API dành cho các hệ điều hành khác nhau là hoàn toàn khác nhau và không có sự tương thích với nhau. API dành cho các hệ điều hành Windows và Linux là hoàn toàn khác nhau.

API cung cấp khả năng truy xuất đến một tập các hàm hay dùng. Web API là một trong những công nghệ mới của Microsoft dùng để xây dựng dịch vụ thành phần phân tán. Web API là mô hình dùng để hỗ trợ MVC bao gồm: routing, controller, action result, filter, lọc container, model binder, unit test, injection. Bên cạnh đó nó còn hỗ trợ đầy đủ các phương thức: Get/Post/put/delete dữ liệu.

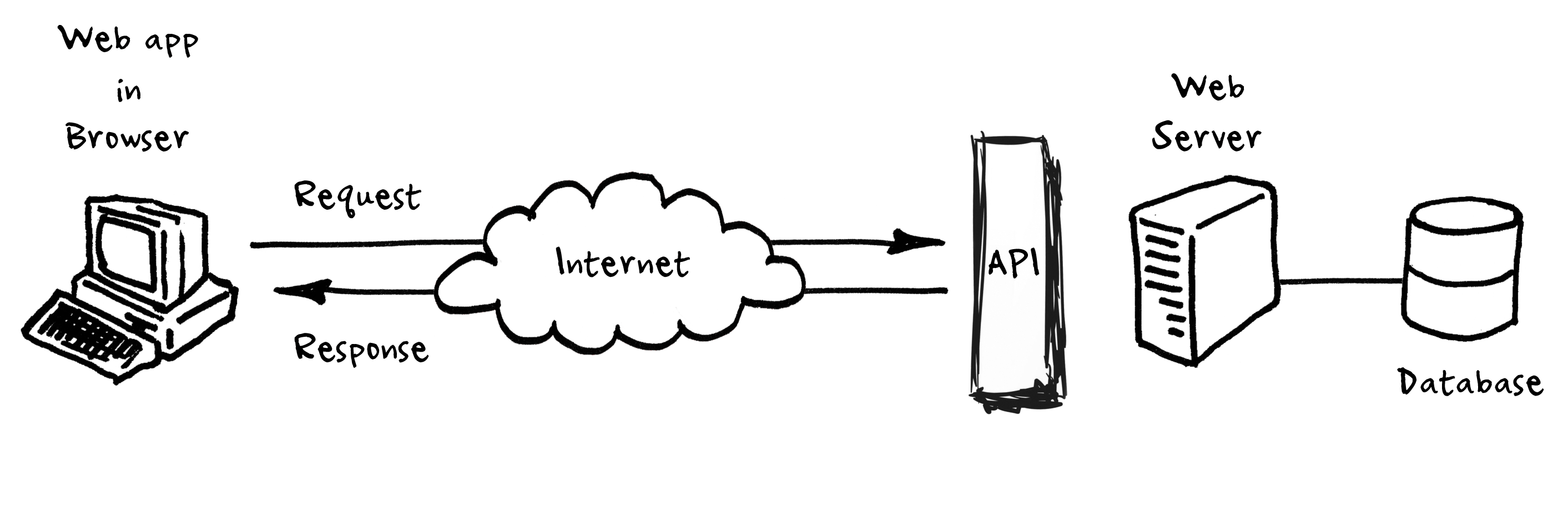
*Những điểm nổi bật của API:*

Đây là một trong những framework mới sẽ giúp ích cho người dùng trong việc xây dựng các HTTP service một cách rất đơn giản và nhanh chóng. Mã nguồn mở nên bạn có thể được sử dụng bởi bất kì một client nào hỗ trợ XML, JSON. Nó cũng có khả năng hỗ trợ đầy đủ các thành phần HTTP: URL, request/response headers,..

* API hiện đại tuân thủ các chuẩn (thường là HTTP và REST), dễ sử dụng, dễ truy cập và dễ hiểu.
* Các API này được xử lý giống như các sản phẩm nhiều hơn là code. Chúng được thiết kế cho các đối tượng người dùng cụ thể (chẳng hạn như các nhà phát triển thiết bị di động), và có các phiên bản cho người dùng và duy trì vòng đời của nó.
* Vì các API được chuẩn hóa nhiều hơn nên vấn đề bảo mật và quản lý cũng nghiêm ngặt hơn, cũng như các vấn đề theo dõi và quản lý hiệu suất, quy mô.
* Giống như các sản phẩm phần mềm khác, API hiện đại cũng có vòng đời phát triển phần mềm riêng từ thiết kế, thử nghiệm, xây dựng, quản lý và các phiên bản. Ngoài ra các API hiện đại cũng được ghi lại cho người dùng và các phiên bản.

### Khái niệm Web API

Web Api là một framework giúp cho việc xây dựng HTTP service một cách dễ dàng. Chúng có thể phát triển cho nhiều clients khác nhau như trình duyệt, mobile app. Web api là một nền tảng để phát triển các ứng dụng dựa trên Restfull service trong .Net được thể hiện dưới hình 1.3.



Hình 1. 4 Mô hình WebAPI

**Web API**hỗ trợ restful đầy đủ các phương thức: Get/Post/put/delete dữ liệu. Rest còn được biết đến là “Representational State Transfer” nó được sử dụng rất nhiều trong công việc phát triển những ứng dụng web services áp dụng giao tiếp HTTP trong giao tiếp thông qua mạng internet. Những ứng dụng sử dụng kiến trúc Rest này còn được gọi là ứng dụng phát triển theo kiểu Restful.

### Ưu điểm của Web API

Vậy ưu điểm của Web API:

* Có được độ hoàn thiện cao, có thể host trong ứng dụng, là kiến trúc lý tưởng cho các thiết bị có băng thông có giới hạn như: Smartphone, tablet.

**Web API service**sử dụng được ở hầu như toàn bộ các client như là ứng dụng desktop, ứng dụng mobile, ứng dụng web.

* **Web API**sẽ trả về client định dữ liệu có thể là Json, Xml hoặc là định dạng khác.
* Xây dựng những HTTP service rất đơn giản nhưng lại vô cùng nhanh chóng.
* Open Source( mã nguồn mở) và có thể được sử dụng với bất kỳ clients nào hỗ trợ Xml, Json.
* Khả năng trình diễn cao.
* Hỗ trợ hoàn toàn đầy đủ HTTP như: URL, request/response headers, content formats, caching, versioning.

## Kết luận

Chương 1 đề cập đến tổng quan đề tài và các vấn đề lý thuyết về cơ sở dữ liệu phân tán, tìm hiểu về Web API và các ưu điểm của Web API khi sử dụng trong hệ thống.

# THIẾT KẾ HỆ THỐNG QUẢN LÝ THIẾT BỊ Y TẾ

Qua quá trình tìm hiểu cơ sở lý thuyết, nghiên cứu và phân tích yêu cầu đề tài, tiến thành phân tích kĩ hệ thống và đưa ra thiết kế các khối chức năng, thiết kế xây dựng cơ sở dữ liệu cho hệ thống.

## Phân tích hệ thống

### Tổng quan hệ thống

Hệ thống quản lý thiết bị y tế bao gồm ba phần đó là WindowApp, Server và Mobile App. Người dùng gửi yêu cầu thông qua Window App hoặc Mobile App, Server sẽ nhận yêu cầu, xử lý và gửi phản hồi về màn hình chính của WinApp và Mobilbe App cho người dùng, được thể hiện như hình 2.1 [3].



Hình 2. 1 Sơ đồ tổng quan hệ thống

### Các đối tượng người dùng

#### Trưởng phòng vật tư

Trưởng phòng vật tư có thể đăng nhập vào hệ thống để:

* Xem thông tin chi tiết các thiết bị trong bệnh viện.
* Nhập mới thiết bị.
* Bàn giao thiết bị cho các đơn vi sử dụng (khoa, phòng).
* Điều chuyển thiết bị cho từng đơn vị sử dụng.

#### Bác sĩ

Cán bộ y tế hay cụ thể là bác sĩ có thể đăng nhập vào hệ thống để quản lý các thiết bị y tế. Cụ thể đó là bác sĩ có thể xem thông tin chi tiết của thiết bị, xem lịch phân công vận hành thiết bị, báo hỏng cho thiết bị và nhận bàn giao thiết bị từ phòng vật tư hay kỹ thuật viên sau khi thiết bị đã được bảo trì , sửa chữa.

#### Kỹ thuật viên

Kỹ thuật viên cũng có thể xem thông tin chi tiết của các thiết bị, nắm bắt được các thiết bị được báo hỏng, thiết bị cần bảo trì để thực hiện bảo trì hay sữa chữa cho thiết bị. Sau khi đã bảo trì xong kỹ thuật viên phải bàn giao lại thiết bị cho đơn vị sử dụng.

### Các yêu cầu chức năng

#### Các chức năng của hệ thống

##### Đăng ký

Đây là bước đầu tiên người dùng phải làm khi cài đặt ứng dụng. Để có thể đăng ký người dùng phải được tạo tên đăng nhập và mật khẩu.

##### Đăng nhập

Sau khi người dùng đã được cấp tài khoản có thể đăng nhập vào hệ thống để sử dụng. Để đăng nhập, người sử dụng phải nhập tài khoản, mật khẩu được cấp trước đó. Nếu đăng nhập trên thiết bị cá nhân thì có thể duy trì đăng nhập để không đăng nhập vào lần sử dụng tiếp theo.

##### Thay đổi mật khẩu

Khi người dùng cần thay đổi mật khẩu đăng nhập, người dùng cần đăng nhập vào hệ thống và tiến hành:

* Nhập mật khẩu mới
* Gửi yêu cầu lên hệ thống

##### Hiển thị thông tin thiết bị

Khi người dùng đăng nhập vào hệ thống, có thể xem thông tin chi tiết từng thiết bị có tại các đơn vị sử dụng hay cả bệnh viện, phòng khám.

##### Báo hỏng thiết bị

Khi có thiết bị cần báo hỏng để yêu cầu sửa chữa, người dùng có quyền báo hỏng sẽ có thể thực hiện thao tác báo hỏng cho thiết bị để các đơn vị liên quan tiếp nhận thông tin và nhận sửa chữa kịp thời.

##### Xác nhận sử dụng thiết bị

Thiết bị sau khi được nhập về hoặc sau khi được sửa chữa sẽ được bàn giao cho đơn vị sử dụng, vì vậy đơn vị liên quan có thể thực hiện thao tác xác nhận sử dụng thiết bị khi đăng nhập vào hệ thống.

#### Nhóm chức năng của Trưởng phòng vật tư

a) Quản lý thiết bị:

Trưởng phòng vật tư có thể thêm , sửa , xóa ,tìm kiếm , xem chi tiết thông tin các thiết bị trong bệnh viện, phòng khám. Cụ thể như tên, số serial, nhà cung cấp, hãng sản xuất, tình trạng thiết bị,…

Ngoài ra trưởng phòng vật tư còn có thể thực hiện các thao tác bàn giao, chuyển tiếp thiết bị đến các đơn vị sử dụng thiết bị trong bệnh viện, phòng khám.

b) Quản lý các đơn vị sử dụng thiết bị

Trưởng phòng vật tư có thể thêm, sửa, xóa các đơn vị sử dụng thiết bị trong bệnh viện hay phòng khám.

Bên cạnh đó, Trưởng phòng vật tư cũng có thể phân quyền, xóa, xem chi tiết, tìm kiếm tài khoản người dùng. Thông tin bao gồm Tên đăng nhập, số điện thoại người dùng.

#### Nhóm chức năng của bác sĩ

a) Quản lý thiết bị

Bác sĩ có thể xem chi tiết, chỉnh sửa tìm kiếm thông tin các thiết bị trong đơn vị sử dụng. Thông tin bao gồm:

* Loại thiết bị
* Số serial
* Nhà cung cấp
* Hãng sản xuất
* Tình trạng thiết bị.

*b) Báo hỏng thiết bị*

Bác sĩ có thể thực hiện thao tác báo hỏng cho thiết bị gặp vấn đề, yêu cầu bảo trì với thiết bị đến hạn bảo trì và xác nhận thiết bị được bàn giao sau khi sửa chữa, bảo trì xong.

##### Báo cáo

Bác sĩ có nhiệm vụ tổng kết và báo cáo tình trạng các thiết bị hiện có, tổng kết số lượng thiết bị được báo hỏng, được bảo trì, được bàn giao.

#### Nhóm chức năng của Kỹ thuật viên

a) Xem thông tin thiết bị

Kỹ thuật viên có thể xem thông tin chi tiết các thiết bị, tiến hành kiểm tra, sữa chữa các thiết bị hỏng, hay cần bảo trì.

b) Bàn giao thiết bị

Sau khi sửa chữa, bảo trì thiết bị, Kỹ thuật viên có trách nhiệm bàn giao tới đơn vị sử dụng thiết bị.

### Các yêu cầu phi chức năng

#### Về hiệu năng

Về mặt tốc độ xử lý, hệ thống cần đảm bảo có tốc độ nhanh, thời gian xử lý yêu cầu nhanh chóng, không gây ra sự khó chịu cho người sử dụng. Trong trường hợp có nhiều người cùng truy cập tại cùng một thời điểm, tốc độ của hệ thống vẫn đạt ở mức chấp nhận được.

Về mặt độ trễ, yêu cầu hệ thống cần có độ trễ thấp.

Về số lượng người dùng, yêu cầu hệ thống cần đáp ứng được số lượng người dùng.

#### Về tính bảo mật,an toàn và khả năng phục hồi

Hệ thống cần phải đảm bảo yếu tố bảo mật tuyệt đối mọi thông tin của người sử dụng.

Có cách thức truy nhập an toàn bằng mật khẩu. Kiểm soát người dùng và phân quyền chặt chẽ, chính xác. Cơ sở dữ liệu được sao lưu và lưu trữ phân tán nhằm phòng tránh các trường hợp bị sai hoặc mất dữ liệu.

Trong các trường hợp bất ngờ hay các sự cố không mong muốn (ví dụ trong trường hợp mất điện, mất kết nối), hệ thống cần có các cơ chế bảo đảm an toàn cho dữ liệu, tránh sai sót thông tin.

Về khả năng phục hồi, hệ thống cần đảm bảo việc dễ phục hồi và sửa chữa dữ liệu nếu có sự cố xảy ra.

Vì hệ thống được thao tác trên môi trường internet nên có thể có rất nhiều vấn đề liên quan đến bảo mật hay sự cố do virus. Do vậy, các yếu tố liên quan đến bảo mật dữ liệu cần phải được coi trọng hơn bao giờ hết.

#### Về tính bảo trì và giao diện

Các chức năng, giao diện đòi hỏi cần phải dễ bảo trì, sửa đổi, bổ sung.

Đối với người dùng, hệ thống cần có giao diện trực quan, thân thiện, dễ sử dụng, dễ thao tác.

Đối với người quản trị hệ thống, giao diện cần trực quan, dễ sử dụng và quản lý.

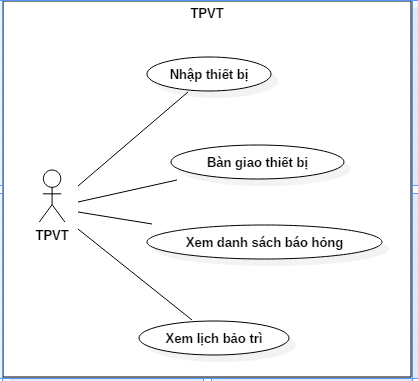
## Thiết kế phần Window App

### Các UseCase trong Window App

#### UseCase TPVT

Trên Window App, Trưởng phòng vật tư có thể thực hiện các thao tác như hình 2.2:

* Nhập thiết bị: TPVT có thể thực hiện nhập thông tin chi tiết của từng thiết bị khi nhập về. Thông tin thiết bị bao gồm:
* Loại thiết bị
* Số Serial
* Hãng sản xuất
* Nhà cung cấp
* Ngày nhập
* Hạn bảo hành
* Chu kì bảo hành
* Bàn giao thiết bị: TPVT có trách nhiệm bàn giao thiết bị tới các đơn vị sử dụng, cụ thể là bàn giao thiết bị cho các bác sĩ tại đơn vị sử dụng
* Xem danh sách báo hỏng: TPVT có thể truy cập vào Window App xem danh sách các thiết bị được đơn vị sử dụng thiết bị báo hỏng.
* Xem lịch bảo trì: TPVT có thể truy cập vào Window App xem danh sách các thiết bị được đơn vị sử dụng thiết bị yêu cầu bảo trì khi đến hạn bảo trì.



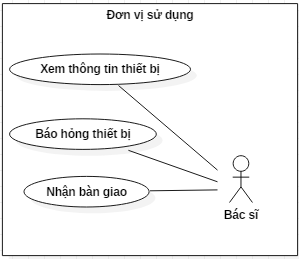
Hình 2. 2 UseCase TPVT trên Window

#### UseCase Bác sĩ

Tại Đơn vị sử dụng, các bác sĩ tại mỗi đơn vị sử dụng có thể dùng WindowApp để thực hiện các thao tác như hình 2.3:

* Xem thông tin thiết bị bao gồm các thông tin:
* Tên
* Số serial
* Loại thiết bị
* Nhà cung cấp
* Hãng sản xuất
* Tình trạng thiết bị
* Báo hỏng thiết bị: khi một thiết bị hỏng hóc, Bác sĩ phải cập nhật trạng tháo “Báo hỏng” cho thiết bị để các đối tượng liên quan có thể biết và tiến hành sửa chữa kịp thời.
* Nhận bàn giao thiết bị: Sau khi thiết bị được bảo trì, sữa chữa xong sẽ được bàn giao về lại đơn vị sử dụng.

Khi đó, Bác sĩ tại đơn vị sử dụng phải tiến hành nhận bàn giao thiết bị đã được sửa chữa hay bảo trì từ đơn vị sửa chữa.



Hình 2. 3 UseCase Bác sĩ trên Window

#### UseCase KTV

Kỹ thuật viên có thể sử dụng Window App để thực hiện các thao tác như trên hình 2.4 bên dưới.

* Xem danh sách báo hỏng:

Khi có thiết bị được báo hỏng, KTV có thể lập tức thấy được thông tin thiết bị để có thể sữa chữa kịp thời.

* Xem lịch bảo trì:

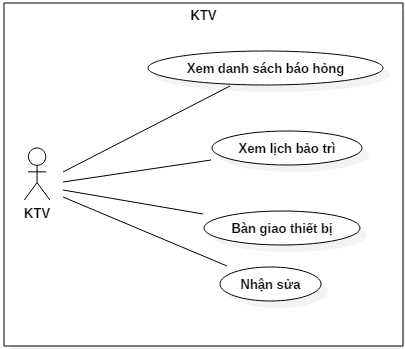
Khi có thiết bị được báo đến hạn bảo trì, KTV có thể lập tức thấy được thông tin thiết bị để có thể tiến hành bảo trì.

* Nhận sửa:

Khi có thiết bị được báo hỏng hay cần bảo trì, KTV có thể tiến hành nhận sửa.

* Bàn giao thiết bị:

Sau khi thiết bị được bảo trì, sữa chữa xong sẽ được bàn giao về lại đơn vị sử dụng.



Hình 2. 4 UseCase KTV trên Window

### Mô tả các hoạt động trên Window App

#### Báo hỏng

##### Báo hỏng

Khi thực hiện thao tác báo hỏng, Bác sĩ sẽ tìm kiếm và chọn tên thiết bị bị hỏng từ danh sách các thiết bị trên Window App. Khi đó, Window App sẽ gọi API báo lỗi rồi gửi lên server, server tìm thiết bị mà bác sĩ đã báo:

* Nếu thiết bị tồn tại, server sẽ thêm ngày giờ vào phần thông tin thiết bị rồi gửi về Window App.

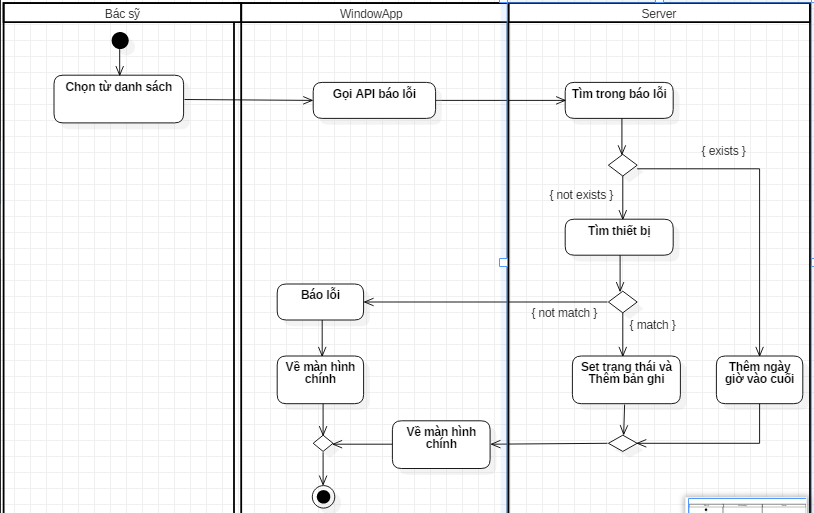
Khi đó Bác sĩ đã hoàn thành việc “Báo hỏng” cho thiết bị, TPVT và KTV sau đấy có thể thấy được thiết bị đã được báo hỏng.

* Nếu thiết bị được báo hỏng không tồn tại, Server tiếp tục tìm kiếm thêm lần nữa.
* Nếu tìm thấy thiết bị, server tiến hành set trạng thái và thêm bản ghi cho thiết bị và quay về màn hình chính ứng dụng.
* Nếu vẫn không tìm thấy thiết bị, server sẽ báo lỗi và trở về màn hình chính ứng dụng.

Khi đó Bác sĩ phải tiến hành kiểm tra xem đã báo hỏng đúng cho thiết bị hay chưa.

Nếu báo sai cần tiến hành thao tác lại từ đầu.

Quá trình báo hỏng được mô tả như trong hình 2.5.



Hình 2. 5 Activity Báo hỏng trên Window

#### Xác nhận sử dụng

Khi thực hiện xác nhận sử dụng, Bác sĩ sẽ tìm kiếm và chọn tên thiết bị cần xác nhận từ danh sách các thiết bị trên phần mềm dành cho Window. Khi đó, WindowApp sẽ gọi API xác nhận sử dụng rồi gửi lên server, Server tìm thiết bị mà bác sĩ đã báo:

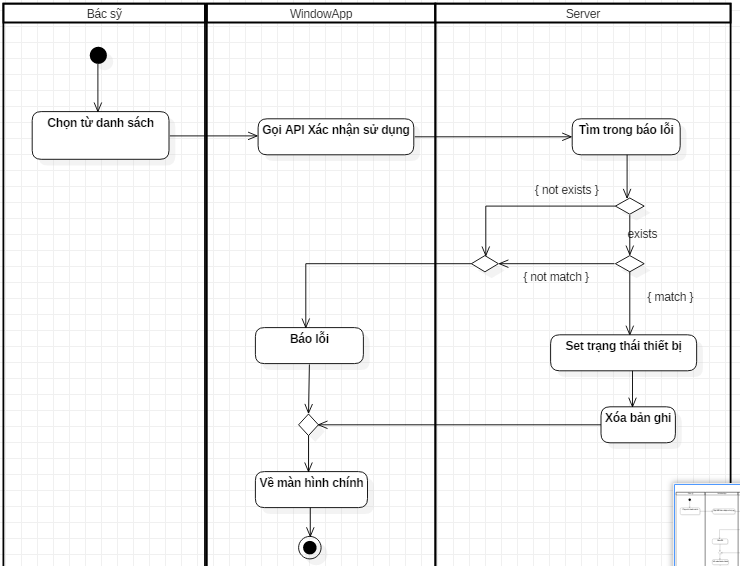
* Nếu thiết bị không tồn tại, server sẽ thực hiện báo lỗi và quay trở về màn hình chính Window App.

Khi đó Bác sĩ đã cần xem lại thông tin thiết bị cần việc “Xác nhận sử dụng”.

* Nếu thiết bị cần xác nhận sử dụng tồn tại, Server tiếp tục tìm kiếm thông tin thiết bị.
* Nếu tìm thấy thiết bị, server tiến hành set trạng thái cho thiết bị, sau đó xóa bản ghi của thiết bi và quay về màn hình chính ứng dụng.
* Nếu vẫn không tìm thấy thiết bị, server sẽ báo lỗi và trở về màn hình chính ứng dụng.

Khi đó Bác sĩ phải tiến hành kiểm tra xem đã tiến hành xác nhận cho đúng thiết bị hay chưa.

Quá trình xác nhận sử dụng được mô tả như trong hình 2.6.



Hình 2. 6 Activity Xác nhận sử dụng trên Window

#### 2.3.2.3 Nhận sửa

Khi thực hiện nhận sửa, Bác sĩ sẽ tìm kiếm và chọn tên thiết bị cần xác nhận từ danh sách các thiết bị trên phần mềm dành cho Window. Khi đó, WindowApp sẽ gọi API xác nhận sửa rồi gửi lên server, Server tìm thiết bị mà bác sĩ đã báo:

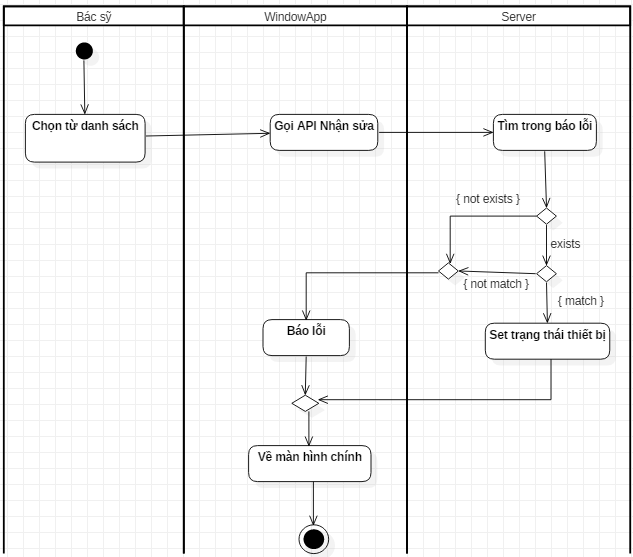
* Nếu thiết bị không tồn tại, server sẽ thực hiện báo lỗi và quay trở về màn hình chính Window App.

Khi đó Bác sĩ đã cần xem lại thông tin thiết bị cần việc “Xác nhận sử dụng”.

* Nếu thiết bị cần nhận sửa tồn tại, Server tiếp tục tìm kiếm thông tin thiết bị.
* Nếu tìm thấy thiết bị, server tiến hành set trạng thái cho thiết bị và quay về màn hình chính ứng dụng.
* Nếu vẫn không tìm thấy thiết bị, server sẽ báo lỗi và trở về màn hình chính ứng dụng.

Khi đó Bác sĩ phải tiến hành kiểm tra xem đã tiến hành xác nhận cho đúng thiết bị hay chưa.

Quá trình xác nhận sử dụng được mô tả như trong hình 2.7.



Hình 2. 7 Activity Nhận sửa trên Window

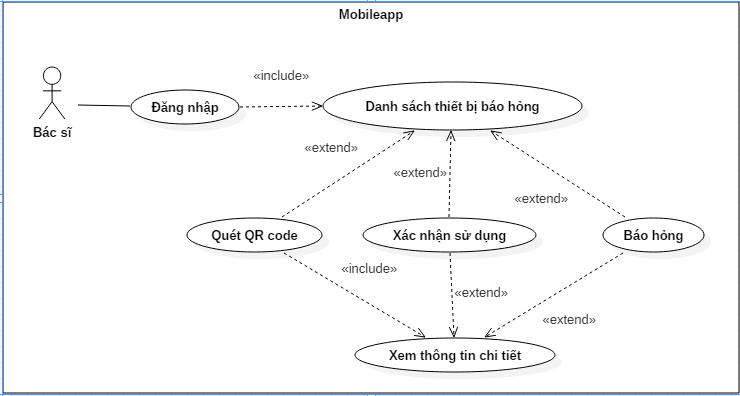
## Thiết kế phần Mobile App

### Các UseCase trong Mobile App

#### UseCase Bác sĩ

Bác sĩ có thể đăng nhập vào Mobile App để thực hiện như hình 2.8:

* Xem danh sách các thiết bị.
* Báo hỏng các thiết bị.
* Quét QR code để xem thông tin thiết bị.
* Mỗi thiết bị sẽ được gắn một mã QR code riêng trong phần thông tin thiết bị.
* Người sử dụng chỉ cần sử dụng chức năng quét mã QR code.
* Thông tin chi tiết hơn của thiết bị sẽ được hiển thị.
* Người dùng có thể xem thông tin chi tiết của thiết bị được quét, bao gồm tên, loại thiết bị, số serial, nhà cung cấp, hãng sản xuất.
* Xác nhận sử dụng thiết bị khi được bàn giao ban đầu hoặc sau khi KTV sửa chữa, bảo dưỡng xong.

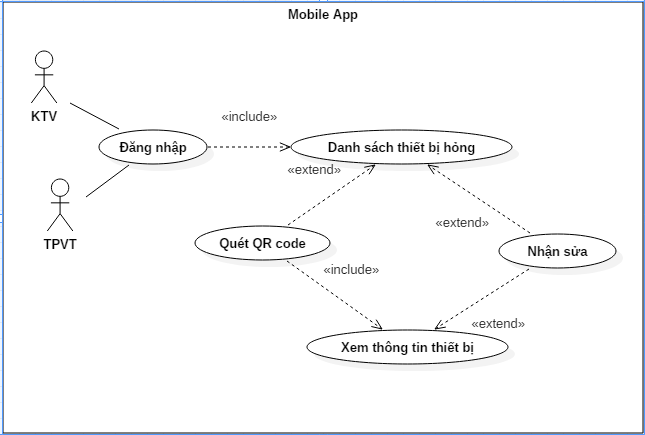


Hình 2. 8 UseCase Bác sĩ trên Mobile App

#### UseCase TPVT và KTV

TPVT và KTV có thể đăng nhập vào Mobile App để:

* Xem thông tin các thiết bị.
* Xem danh sách các thiết bị báo hỏng.
* Nhận sửa các thiết bị.
* Quét QR code để xem thông tin thiết bị:
* Mỗi thiết bị sẽ được gắn một mã QR code riêng trong phần thông tin thiết bị tại mỗi đơn vị.
* Người sử dụng chỉ cần sử dụng chức năng quét mã QR code.
* Thông tin chi tiết hơn của thiết bị sẽ được hiển thị.
* Người dùng có thể xem thông tin chi tiết của thiết bị được quét, bao gồm tên, loại thiết bị, số serial, nhà cung cấp, hãng sản xuất.
* Nếu thiết bị đã được báo hỏng, KTV có thể tiến hành nhận sửa với thiết bị đó.



Hình 2. 9 UseCase TPVT và KTV trên Mobile App

### Mô tả các hoạt động trên Mobile App

#### Báo hỏng

Khi thực hiện thao tác báo hỏng, Bác sĩ sẽ dùng chức năng quét mã QR code hoặc có thể tìm kiếm và chọn tên thiết bị bị hỏng từ danh sách các thiết bị trên phần mềm trên điện thoại. Khi đó, MobileApp sẽ gọi API báo lỗi rồi gửi lên server, server tìm thiết bị mà bác sĩ đã báo:

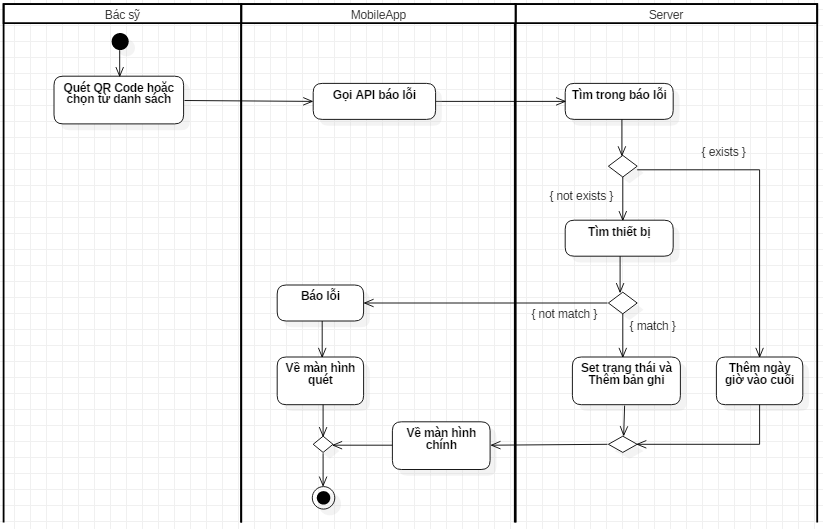
* Nếu thiết bị tồn tại, server sẽ thêm ngày giờ vào phần thông tin thiết bị rồi gửi về điện thoại.

Khi đó Bác sĩ đã hoàn thành việc “Báo hỏng” cho thiết bị, TPVT và KTV sau đấy có thể thấy được thiết bị đã được báo hỏng.

* Nếu thiết bị được báo hỏng không tồn tại, Server tiếp tục tìm kiếm thêm lần nữa.
* Nếu tìm thấy thiết bị, server tiến hành set trạng thái cho thiết bị và quay về màn hình chính ứng dụng
* Nếu vẫn không tìm thấy thiết bị, server sẽ báo lỗi và trở về màn hình quét của điện thoại người dùng.

Khi đó Bác sĩ phải tiến hành kiểm tra xem đã báo hỏng đúng cho thiết bị hay chưa.

Quá trình báo hỏng được mô tả như trong hình 2.10.



Hình 2. 10 Activity Báo hỏng trên Mobile App

#### Xác nhận sử dụng

Khi thực hiện xác nhận sử dụng, Bác sĩ sẽ dùng chức năng quét mã QR code hoặc có thể tìm kiếm và chọn tên thiết bị cần xác nhận từ danh sách các thiết bị trên phần mềm dành cho điện thoại. Khi đó, MobileApp sẽ gọi API xác nhận sử dụng rồi gửi lên server, server tìm thiết bị mà bác sĩ đã báo.

Khi đó xảy ra hai trường hợp:

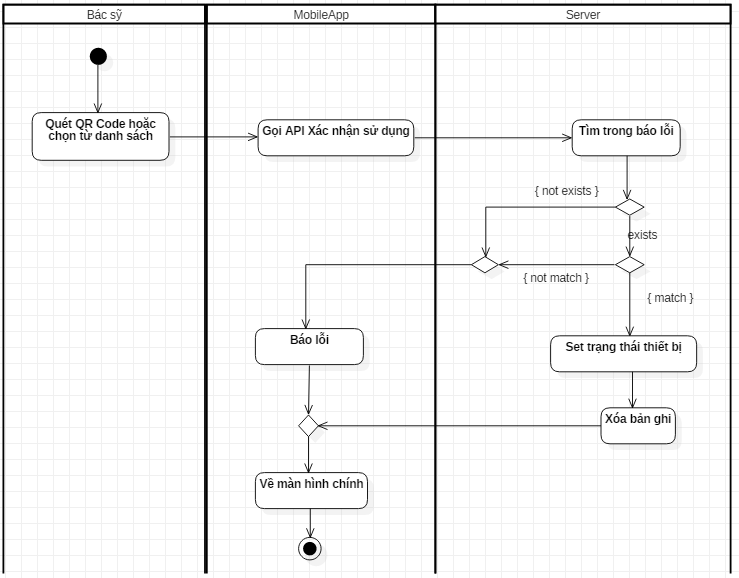
* Nếu thiết bị không tồn tại, server sẽ thực hiện báo lỗi rồi gửi về màn hình chính Mobile App.

Khi đó Bác sĩ cần xem lại việc “Xác nhận sử dụng” cho thiết bị đúng hay chưa.

* Nếu thiết bị cần xác nhận tồn tại, Server tiếp tục tìm kiếm thiết bị.
* Nếu tìm thấy thiết bị, server tiến hành set trạng thái cho thiết bị và xóa bản ghi rồi quay về màn hình chính ứng dụng
* Nếu vẫn không tìm thấy thiết bị, server sẽ báo lỗi và trở về màn hình quét của điện thoại người dùng.

Khi đó Bác sĩ phải tiến hành kiểm tra xem đã xác nhận đúng cho thiết bị hay chưa.

Quá trình xác nhận sử dụng được mô tả như trong hình 2.11.



Hình 2. 11 Activity Xác nhận sử dụng trên Mobile App

#### Nhận sửa

Khi thực hiện nhận sửa, Bác sĩ sẽ sử dụng chức năng quét mã QR code hoặc có thể tìm kiếm và chọn tên thiết bị cần xác nhận từ danh sách các thiết bị trên phần mềm dành cho điện thoại. Khi đó, MobileApp sẽ gọi API xác nhận sửa rồi gửi lên server, Server tìm thiết bị mà bác sĩ đã báo:

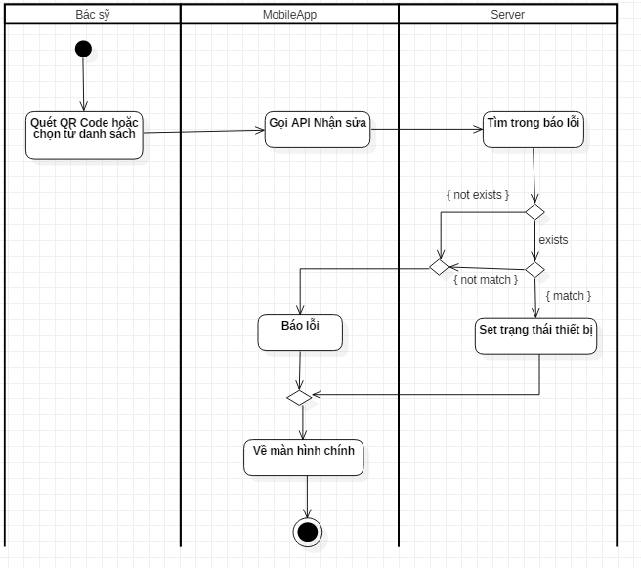
* Nếu thiết bị không tồn tại, server sẽ thực hiện báo lỗi rồi gửi về màn hình chính Mobile App.

Khi đó Bác sĩ đã hoàn thành việc “Nhận sửa” cho thiết bị.

* Nếu thiết bị cần xác nhận tồn tại, Server tiếp tục tìm kiếm thiết bị.
* Nếu tìm thấy thiết bị, server tiến hành set trạng thái cho thiết bị và quay về màn hình chính ứng dụng.
* Nếu vẫn không tìm thấy thiết bị, server sẽ báo lỗi và trở về màn hình chính của điện thoại người dùng.

Khi đó Bác sĩ phải tiến hành kiểm tra xem đã xác nhận sửa cho đúng thiết bị hay chưa.

Quá trình xác nhận sử dụng được mô tả như trong hình 2.12.



Hình 2. 12 Activity Nhận sửa trên Mobile

## Cơ sở dữ liệu MongoDB

### Khái niệm MongoDB và NoSQL

NoSQL nghĩa là Non SQL hay cũng có thể gọi là Not only SQL [11].NoSQL là cơ sở dữ liệu không quan hệ, ràng buộc giữa các Collection (tương đương với table trong CSDL thông thường). NoSQL đặc biệt nhấn mạnh đến mô hình lưu trữ cặp key – value và hệ thống lưu trữ phân tán.

NoSQL là mã nguồn mở, không cần mất chi phí và có xu hướng tin cậy, nhanh hơn so với hệ quản tri cơ sở dữ liệu quan hệ [12]. NoSQL linh hoạt trong việc mở rộng và phát triển, đồng thời áp dụng được công nghệ điện toán đám mây, vì vậy đem lại lợi ích đáng kể cho người dùng.

MongoDB là một hệ quản trị cơ sở dữ liệu mã nguồn mở thuộc họ NoSQL. MongoDB có cấu trúc lưu trữ tương tự như JSON, hoạt động trên khái niệm collection và document.

**Collection trong MongoDB** là nhóm các tài liệu (**document**), nó tương đương với một bảng (table) trong CSDL thông thường nên mỗi collection sẽ thuộc về một database duy nhất. Tuy nhiên nó có một sực khác biệt đó là nó không có ràng buộc Relationship như các hệ quản trị CSDL khác nên việc truy xuất rất nhanh, chính vì thế mỗi collection có thể chứa nhiều thể loại khác nhau không giống như table trong hệ quản trị mysql là các field cố định [13].

Document trong MongoDB có cấu trúc tương tự như kiểu dữ liệu JSON, nghĩa là sẽ có các cặp (key- giá trị) nên nó có tính năng động rất lớn.

Document ta có thể hiểu nó giống như các record dữ liệu trong MYSQL, tuy nhiên nó có sự khác biệt là các cặp (key- value) có thể không giống nhau ở mỗi document. Bảng 2.1 cho thấy sự khác nhau giữa MongoDB và SQL Database.

Bảng 2. 1 So sánh SQL Database và MongoDB

| **SQL DB** | **MongoDB** |
| --- | --- |
| Table | Collection |
| Row | Document |
| Column | Field |
| Joins | Embeded documents, linking |
| Primary key | Primary key |

MongoDB là một cơ sở dữ liệu NoSQL hỗ trợ đa nền tảng, nó có thể chạy trên Windows, Linux và Mac...Nó hỗ trợ hầu hết các ngôn ngữ lập trình phổ biến như C#, Java, PHP, Javascript...và các môi trường phát triển khác nhau.

### Đặc điểm của MongoDB

MongoDB có một số đặc điểm như:

* MongoDB lưu dữ liệu dạng JSON, khi bạn insert nhiều đối tượng thì nó sẽ là insert một mảng JSON gần như với trường hợp insert 1 đối tượng.
* Dữ liệu trong MongoDB không có sự ràng buộc lẫn nhau như trong RDBMS, khi insert, xóa hay update nó không cần phải mất thời gian kiểm tra xem có thỏa mãn các bảng liên quan như trong RDBMS.
* Dữ liệu trong MongoDB được [đánh chỉ mục](https://stackjava.com/mongodb/danh-chi-muc-la-gi-danh-chi-muc-trong-mongodb.html) ([đánh index](https://stackjava.com/mongodb/danh-chi-muc-la-gi-danh-chi-muc-trong-mongodb.html)) nên khi truy vấn sẽ tìm rất nhanh..
* Phi quan hệ - không có ràng buôc nào cho việc nhất quán dữ liệu
* Tính nhất quán không theo thời gian thực: sau mỗi thay đổi CSDL, không cần tác động ngay đến tất cả các CSDL liên quan mà được lan truyền theo thời gian.

### Kiến trúc MongoDB

Một MongoDB Server sẽ chứa nhiều database. Mỗi database lại chứa một hoặc nhiều colection. Đây là một tập các documnents, về mặt logic thì chúng gần tương tự như các table trong CSDL quan hệ. Tuy nhiên, điểm hay ở đây là ta không cần phải định nghĩa trước cấu trúc của dữ liệu trước khi thao tác thêm, sửa dữ liệu… Một document là một đơn vị dữ liệu – một bản ghi (không lớn hơn 16MB). Mỗi chúng lại chứa một tập các trước hoặc các cặp key – value. Key là một chuỗi ký tự, dùng để truy xuất giá trị dạng : string, integer, double, …

Cấu trúc có vẻ khá giống JSON, tuy nhiên, khi lưu trữ document này ra database, MongoDB sẽ serialize dữ liệu thành một dạng mã hóa nhị phân đặc biệt – BSON. Ưu điểm của BSON là hiệu quả hơn các dạng format trung gian như XML hay JSON cả hệ tiêu thụ bộ nhớ lẫn hiệu năng xử lý. BSON hỗ trợ toàn bộ dạng dữ liệu mà JSON hỗ trợ (string, integer, double, Boolean, array, object, null) và thêm một số dạng dữ liệu đặc biệt như regular expression, object ID, date, binary, code.

### Ưu điểm của việc sử dụng cơ sở dữ liệu MongoDB

* Dễ học, có một số nét khá giống với CSDL quan hệ, được quản lý bằng command line hoặc bằng GUI như RockMongo.
* Linh động, không cần phải định nghĩa cấu trúc dữ liệu trước khi tiến hành lưu trữ nó.
* Khả năng mở rộng tốt, khả năng cân bằng tải cao, tích hợp các công nghệ quản lý dữ liệu vẫn tốt khi kích thước và thông lượng trao đổi dữ liệu tăng.
* Schema linh hoạt: Do MongoDB sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ các các kích cỡ và các document khác nhau.
* Cấu trúc đối tượng rõ ràng: Tuy rằng cấu trúc của dữ liệu là linh hoạt nhưng đối tượng của nó được xác định rất rõ ràng. Sử dụng bộ nhớ nội tại, nên truy vấn sẽ rất nhanh.
* Không có các join: Điều này cũng góp phần tạo nên tốc độ truy vấn cực nhanh trên mongoDB.
* MongoDB rất dễ mở rộng.
* MongoDB phù hợp cho các ứng dụng realtime.

## Kết luận

Chương 2 đã trình bày các bước phân tích hệ thống, đưa ra thiết kế cho từng phần Server, Window App và Mobile App, từ đó tiến hành xây dựng hệ thống quản lý thiết bị y tế hoàn chỉnh nhất.

# XÂY DỰNG BACKEND APPLICATION

Sau khi tìm hiểu các cơ sở lý thuyết ở chương 1, phân tích và thiết kế hệ thống, xây dựng cơ sở dữ liệu ở chương 2, ở chương 3 tiến hành xây dựng bộ API dùng cho server và kiểm nghiệm, đánh giá kết quả đạt được.

## Xây dựng bộ API trên Server

### API Acount

API của lớp Account bao gồm các chức năng như bảng 3.1:

* Chức năng đăng kí (Register): Dữ liệu đầu vào là tên người dùng (UserName), kiểu dữ liệu string.
* Chức năng thay đổi mật khẩu (ChangePassword): Dữ liệu đầu vào là mật khẩu mới (newPassword), kiểu dữ liệu string.
* Chức năng đăng nhập (Login): Dữ liệu đầu vào là tên người dùng (UserName) và mật khẩu (Pasword), kiểu dữ liệu string.
* Chức năng thoát đăng nhập (Logout).

Bảng 3. 1 API Account



### API Thiết bị

API lớp Thiết bị bao gồm các chức năng như bảng 3.2:

* Thêm các thiết bị (Register): Dữ liệu đầu vào là (objectId), kiểu dữ liệu là string.
* Xem danh sách các thiết bị (List).
* Xem thông tin các thiết bị (Detail).
* Xóa các thiết bị (Move).
* Quản lý các thiết bị (Manager).
* Quét mã QR (QR).
* Xem lịch sử thiết bị (History).

Bảng 3. 2 API Thiết bị



### API Bảo trì

API lớp Bảo trì bao gồm các chức năng như bảng 3.3:

* Xác nhận bảo trì thiết bị (Support).
* Xem danh sách thiết bị cần bảo trì (List).
* Xem lịch các thiết bị được bảo trì theo từng tháng (Calendar).
* Xác nhận đã bảo trì (Done).
* Lấy phiếu bảo trì (Coupon).
* Viết phiếu bảo trì (Write).

Bảng 3. 3 API Bảo trì



### API Danh mục

API lớp Danh mục bao gồm các chức năng như bảng 3.4:

* Phân loại các thiết bị (Phanloai): người dùng chỉ cần đăng nhập vào hệ thống và chọn phân loại thiết bị, kết quả trả về các loại thiết bị y tế mà bệnh viện có.
* Tên các nhà sản xuất thiết bị (HangSX): người dùng chỉ cần đăng nhập vào hệ thống và chọn Danh mục hãng sản xuất thiết bị, kết quả trả về các hãng sản xuất thiết bị y tế mà bệnh viện đã nhập.
* Tên các nhà cung cấp thiết bị (NhaCC): người dùng chỉ cần đăng nhập vào hệ thống và chọn Danh mục nhà cung cấp thiết bị, kết quả trả về các nhà cung cấp thiết bị y tế mà bệnh viện đã nhập.
* Chỉnh sửa thông tin thiết bị (EditTB): người dùng chỉ cần đăng nhập vào hệ thống và chọn Edit thiết bị mong muốn, kết quả trả về thông tin thiết bị sau khi được chỉnh sửa.

Bảng 3. 4 API Danh mục



### API Báo hỏng

API lớp Báo hỏng bao gồm các chức năng như bảng 3.5:

* Báo hỏng cho thiết bị (Register)
* Xác nhận thiết bị đã được sửa chữa xong(Accept)
* Xem danh sách các thiết bị đã báo hỏng (List)
* Thông báo có thiết bị hỏng (Notification)
* Sửa chữa các thiết bị được báo hỏng (Support)

Bảng 3. 5 API Báo hỏng



### API Khách hàng

API lớp Khách hàng bao gồm các chức năng như bảng 3.6:

* Thêm khách hàng (Register)
* Xem danh sách các khách hàng (List)

Bảng 3. 6 API Khách hàng



### API Database

API lớp Database bao gồm các chức năng như bảng 3.7:

* Đặt lại dữ liệu (Reset)

Bảng 3. 7 API Database



### API Đơn vị

API lớp Đơn vị bao gồm các chức năng như bảng 3.8:

* Thêm đơn vị sử dụng (Register)
* Xem danh sách các đơn vị sử dụng (List)

Bảng 3. 8 API Đơn vị



### API User

API lớp User bao gồm các chức năng như bảng 3.9:

* Tìm kiếm người dùng (Search)
* Đặt lại tài khoản người dùng (Reset)

Bảng 3. 9 API User



### API Xử lý

API lớp Xử lý bao gồm các chức năng như bảng 3.10:

* Xem danh sách các thiết bị được xử lý (List)

Bảng 3. 10 API Xử lý



### API Báo cáo

API lớp Báo cáo bao gồm các chức năng như bảng 3.11:

* Báo cáo theo từng đơn vị.

Các đơn vị sử dụng sẽ báo cáo số lượng các thiết bị hiện có theo từng đơn vị (Devices).

* Báo cáo theo toàn bệnh viện.

TPVT báo cáo số lượng thiết bị của toàn viện (Department).

Bảng 3. 11 API Báo cáo



## Kiểm thử

### Giao diện trên Web

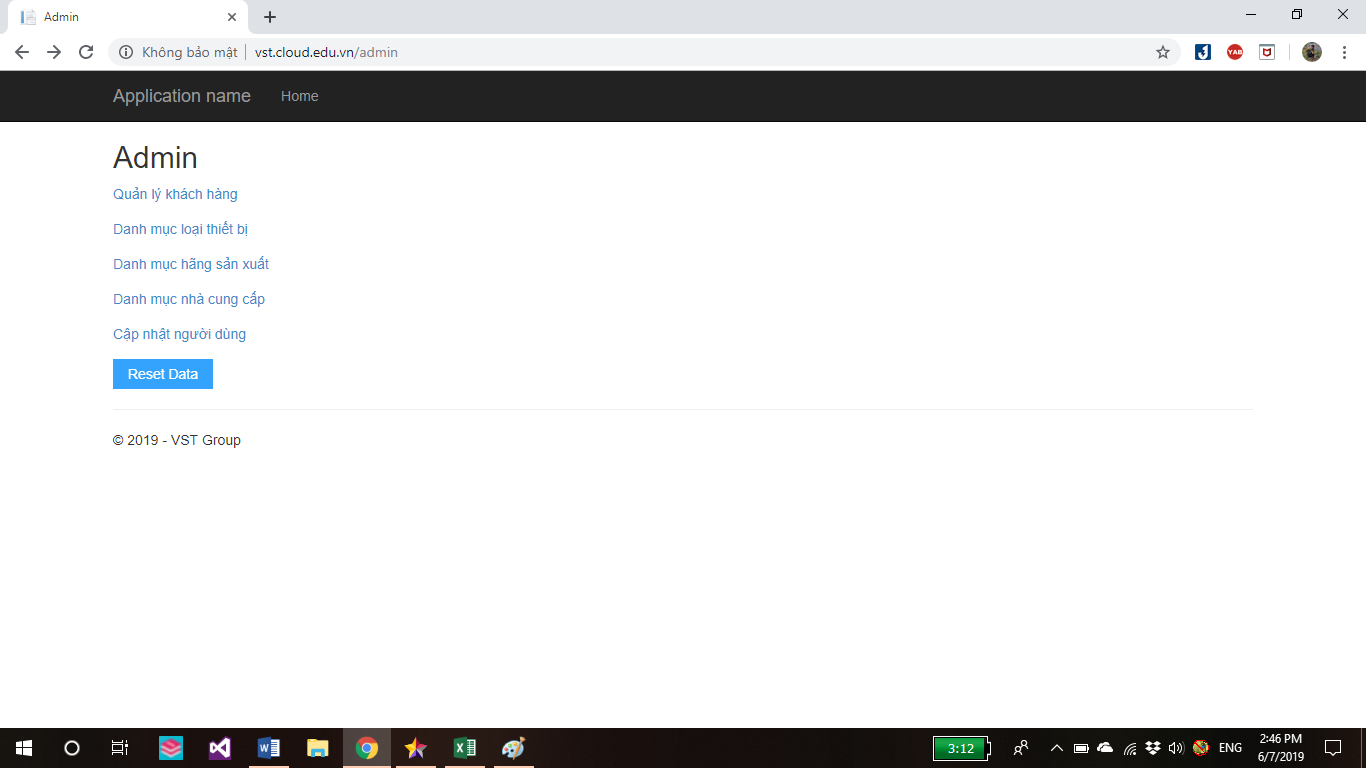
Giao diện trang web dành cho admin được thể hiện dưới hình 3.1.

Người quản lý có thể thực hiện các chức năng:

* Quản lý khách hàng
* Quản lý thiết bị
* Xem danh mục các hãng sản xuất
* Xem danh mục các nhà cung cấp
* Cập nhật người dùng

Khi người dùng muốn quay trở về màn hình chính thì chọn Home.

Còn khi người dùng muốn đăng xuất khỏi hệ thống thì chọn Application name.



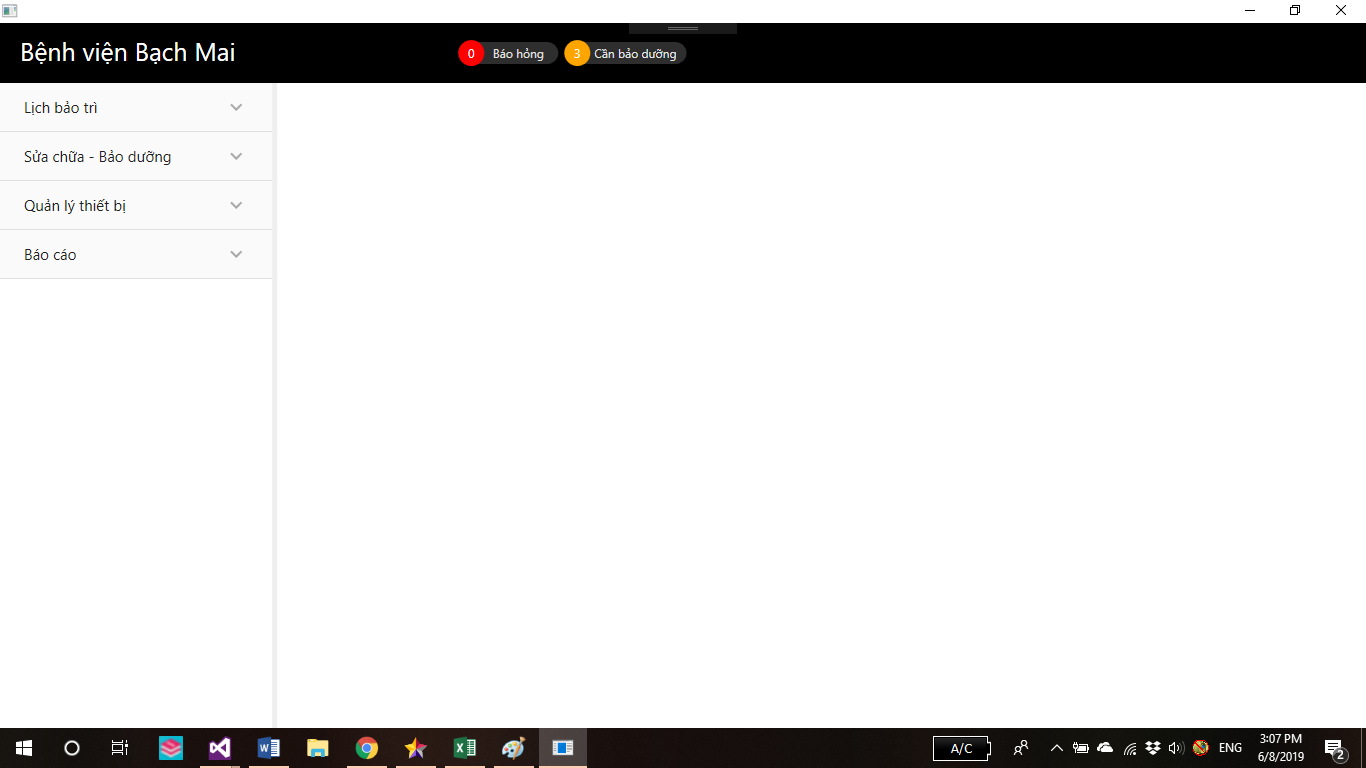
Hình 3. 1 Giao diện Web quản lý thiết bị

### Giao diện trên Window

Giao diện WindowApp được thể hiện dưới hình 3.2.

Người sử dụng có thể xem các thông tin:

* Số lượng thiết bị báo hỏng
* Số lượng thiết bị cần bảo dưỡng
* Lịch bảo trì
* Thông tin sửa chữa – bảo dưỡng
* Báo cáo



Hình 3. 2 Giao diện quản lý thiết bị trên Window

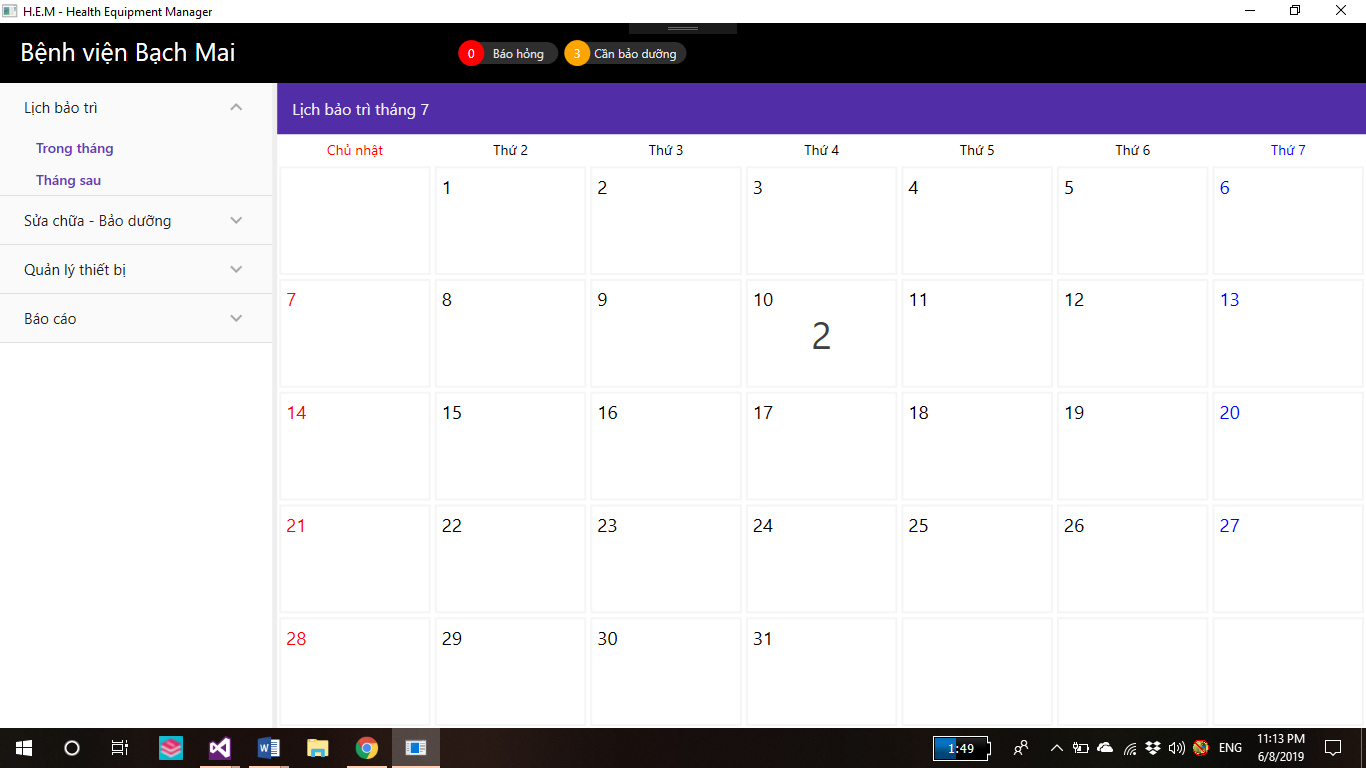
Người sử dụng có thể xem lịch bảo trì theo hai tháng liên tiếp như hình 3.3.

Số lượng thiết bị cần bảo trì sẽ hiện lên theo từng ngày trong tháng.

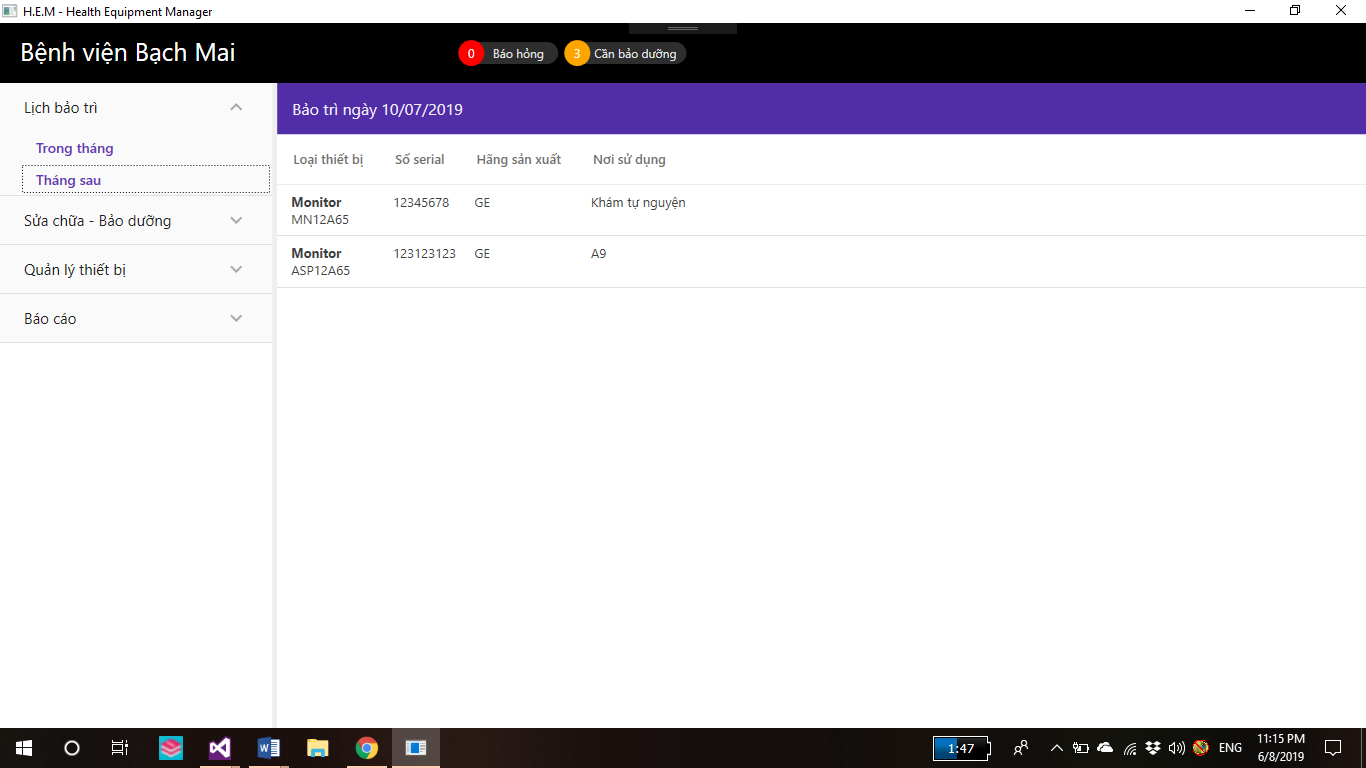
Khi nhấn chuột vào ngày có số lượng thiết bị cần bảo trì sẽ hiển thị thông tin chi tiết thiết bị như hình 3.4.

Thông tin chi tiết thiết bị bao gồm:

* Ngày tháng bảo trì
* Tên thiết bị
* Loại thiết bị
* Số serial
* Hãng sản xuất
* Đơn vị sử dụng



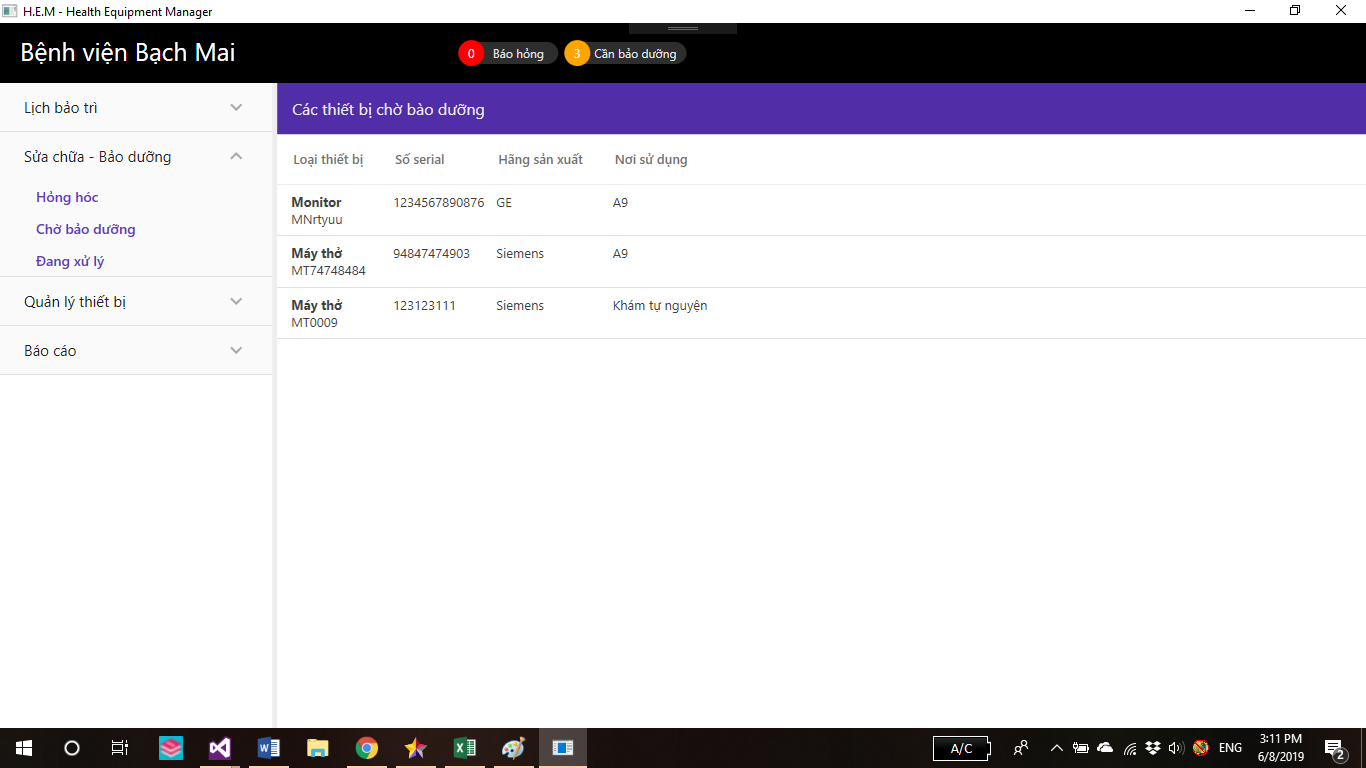
Hình 3. 3 Giao diện lịch bảo trì trên Window



Hình 3. 4 Giao diện thông tin thiết bị cần bảo trì trên Window App

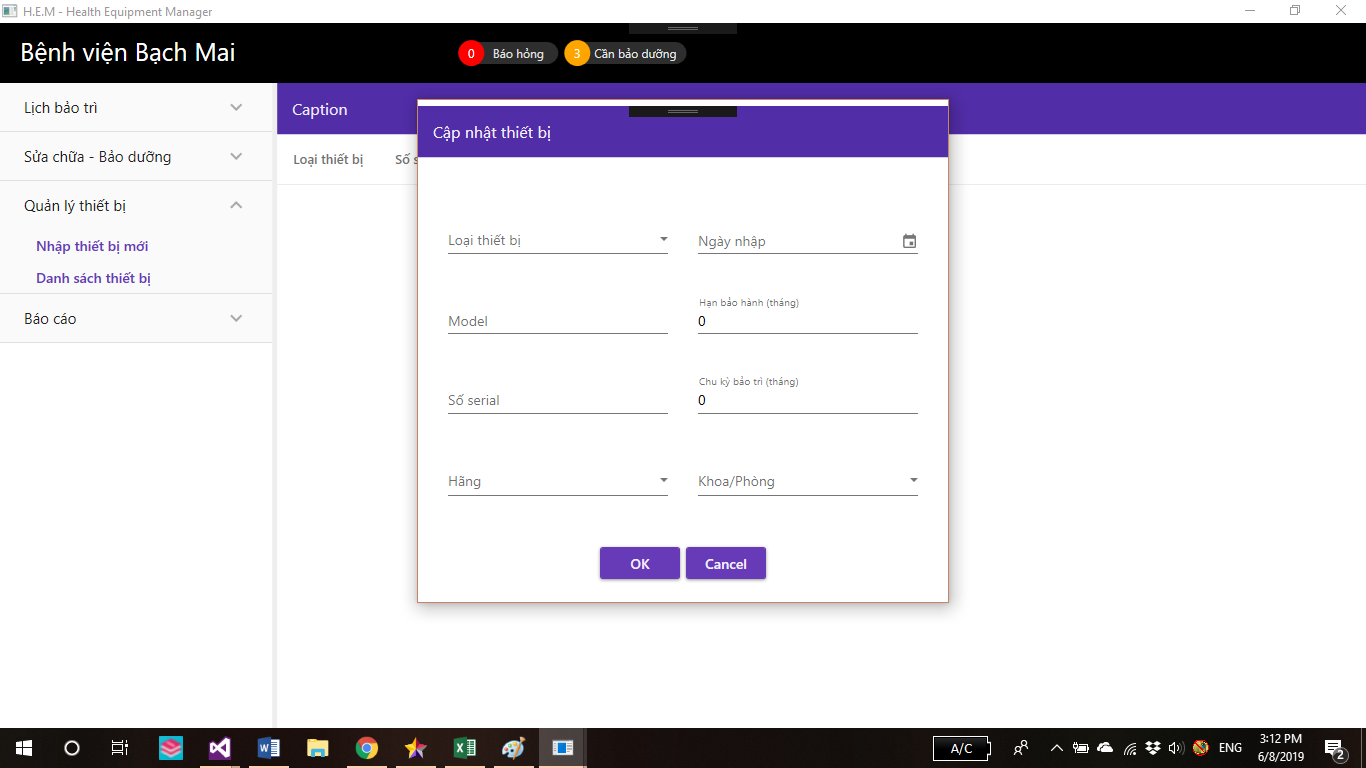
Ở chức năng sửa chữa – bảo dưỡng sẽ hiển thị danh sách các thiết bị hỏng hóc, chờ bảo dưỡng hay đang được xử lý như hình 3.5. Thông tin chi tiết thiết bị bao gồm:

* Ngày tháng bảo trì
* Loại thiết bị
* Số serial
* Hãng sản xuất
* Đơn vị sử dụng



Hình 3. 5 Giao diện danh sách các thiết bị chờ bảo dưỡng trên Window App

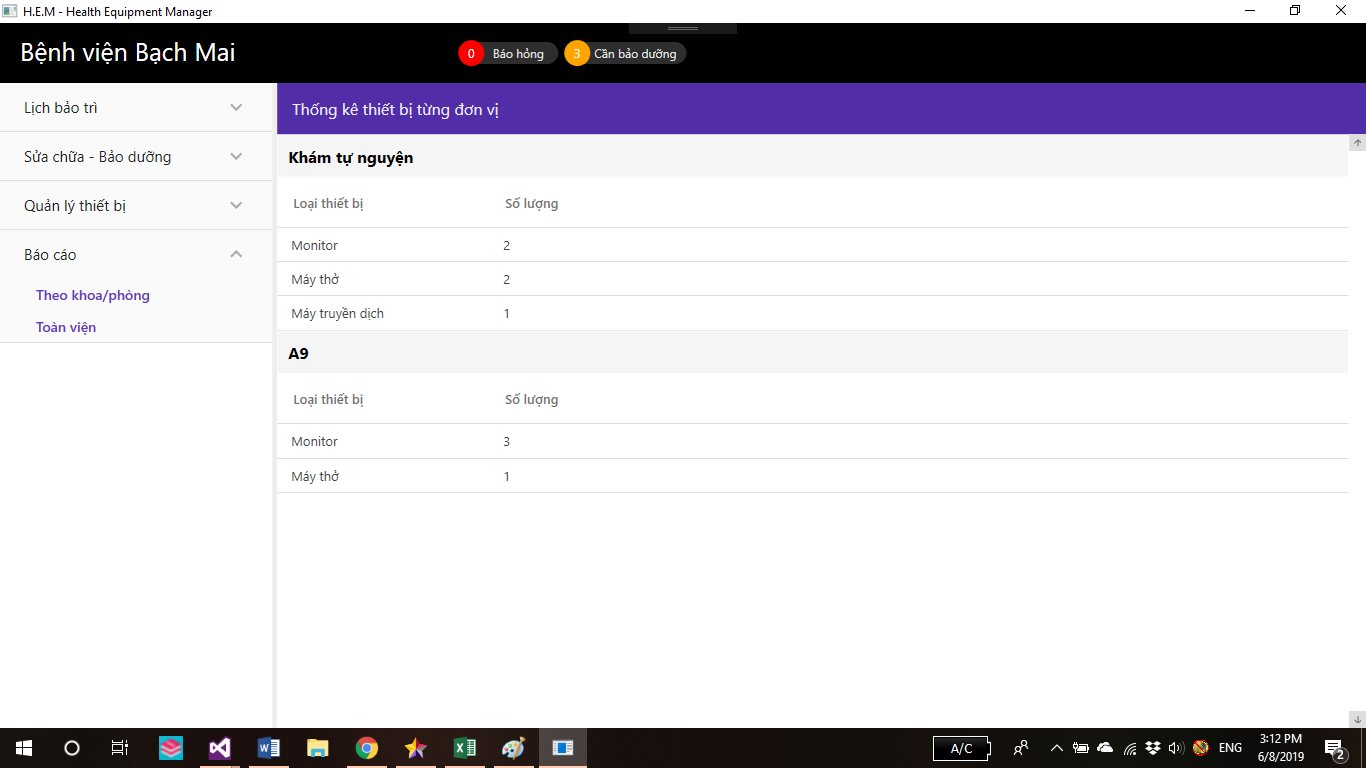
Khi nhập mới thiết bị, cần nhập đầy đủ thông tin chi tiết của thiết bị mới nhập như hình 3.6. Bao gồm các thông số như: Loại thiết bị, Model, Số serial, Hãng, Ngày nhập, Hạn bảo hành, Chu kì bảo trì, và Đơn vị sử dụng thiết bị. Thiết bị sau khi được cập nhật thông tin sẽ hiển thị trên màn hình chính phần Quản lý thiết bị.



Hình 3. 6 Giao diện màn hình cập nhật thiết bị trên Window App

Tại mục Báo cáo sẽ hiển thị danh sách thống kê số lượng các thiết bị đang có theo từng đơn vị sử dụng, hay theo toàn bệnh viện.

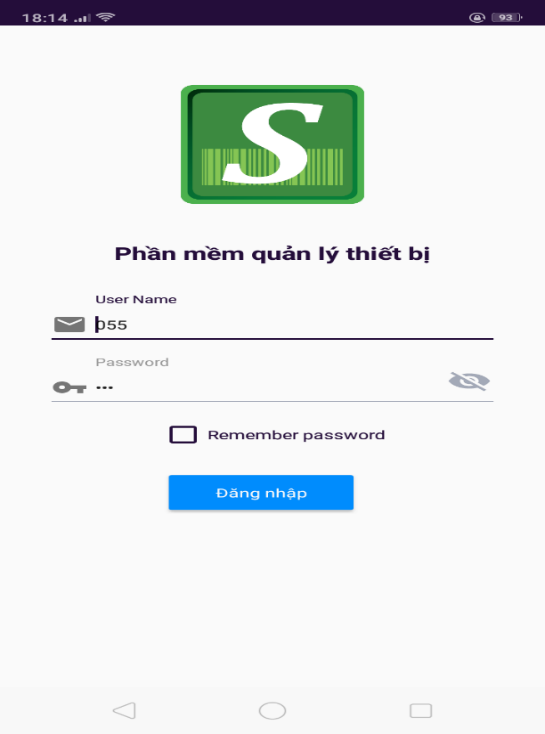
Thống kê các thiết bị được hiển thị dưới hình 3.7.



Hình 3. 7 Giao diện phần thống kê thiết bị từng đơn vị trên Window App

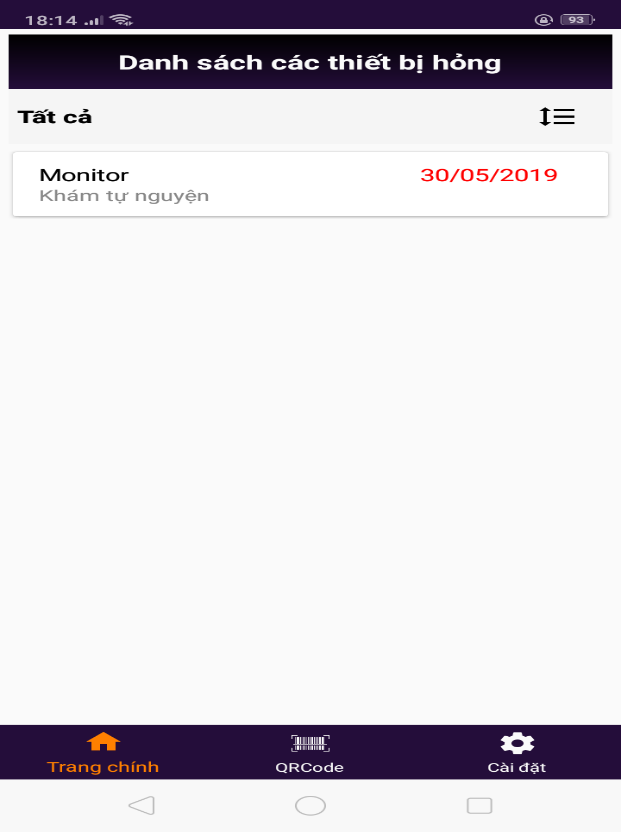
### Giao diện trên Mobile App

Giao diện màn hình đăng nhập vào Mobile App được thể hiện như hình 3.8.Người dùng cần nhập đúng tên đăng nhập (User Name) và mật khẩu (Pass Word) để truy cập vào ứng dụng.“Remember password“ là chế độ duy trì đăng nhập, ứng dụng sẽ lưu tên và mật khẩu đã đăng nhập để sử dụng cho những lần đăng nhập tiếp theo.

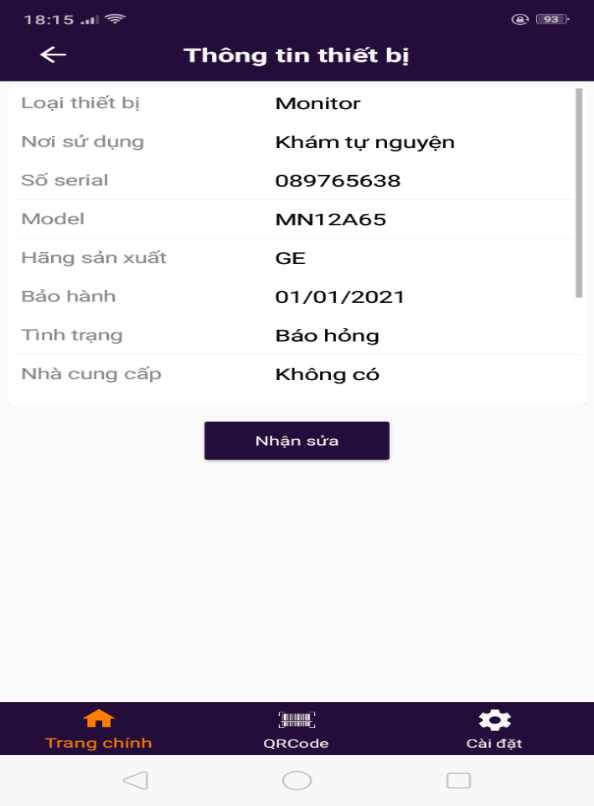


Hình 3. 8 Giao diện màn hình đăng nhập trên Mobile App

Sau khi đăng nhập vào ứng dụng, các đối tượng liên quan là TPVT và KTV có thể xem danh sách các thiết đã được báo hỏng để có thể tiến hành nhận sửa giống như hình 3.8 bên dưới. Thông tin bao gồm loại thiết bị, đơn vị sử dụng và ngày tháng được báo hỏng. Chọn vào phần thông tin thiết bi sẽ hiên ra thông tin chi tiết của thiết bị được chọn.

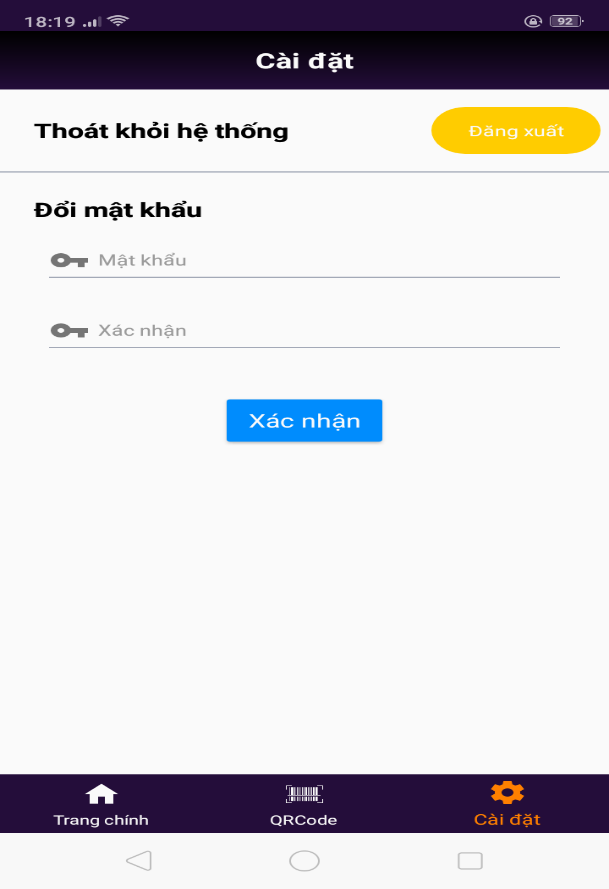


Hình 3. 9 Danh sách các thiết bị hỏng trên Mobile App



Hình 3. 10 Màn hình Mobile App sau khi quét mã QR code

Mobile App còn có chức năng quét mã QR code, mỗi thiết bị được gắn một mã QR, người dùng có thể sử dung chức năng quét mã QR code đó, màn hình ngay lập tức sẽ hiển thị thông tin chi tiết của thiết bị được quét. Đối với thiết bị đã được báo hỏng, KTV có thể tiến hành nhận sửa như trên hình 3.10.



Hình 3. 11 Giao diện đăng xuất và thay đổi mật khẩu trên Mobile App

Khi cần thay đổi mật khẩu hoặc đăng xuất khỏi ứng dụng, người dùng chỉ cần chọn mục Cài đặt như trên hình 2.23 và :

* Chọn Đăng xuất khi muốn thoát khỏi ứng dụng.

Nhập mật khẩu mới và xác nhận mật khẩu mới khi muốn thay đổi mật khẩu và gửi lên hệ thống, khi đó mật khẩu của người dùng đã được thay đổi.

## Đánh giá kết quả

Hệ thống quản lý thiết bị y tế bao gồm ba phần Server, Window App và Mobile App ban đầu đã đáp ứng được các yêu cầu đặt ra, dễ dàng sử dụng với mọi đối tượng và đem lại hiệu quả nhất định trong quản lý thiết bị y tế.

## Kết luận

Chương 3 đã trình bày quá trình xây dưng bộ API trên server và các bước kiểm thử, đánh giá hiệu quả làm việc của từng phần trong hệ thống quản lý thiết bị y tế so với yêu cầu đặt ra.

# KẾT LUẬN

## Kết luận chung

Sau quá trình tìm hiểu, phân tích, thiết kế và triển khai xậy dựng, hệ thống quản lý thiết bị y tế hoạt động trên Window, Server và Mobile App đã cơ bản được hoàn thành. Hệ thống mang lại hiệu quả nhất định trong việc quản lý các thiết bị y tế trong bệnh viện hay các phòng khám.

WindowApp và Mobile App có giao diện thân thiện với người dùng, dễ thực hiện các thao tác quản lý thiết bị. Server chạy ổn định.

Trong thời gian tới, các thành phần của hệ thống được hoàn thiện về chức năng và được tối ưu để đáp ứng với nhu cầu thực tế.

# TÀI LIỆU THAM KHẢO

1. *Cơ sở dữ liệu phân tán* – Học Viện Bưu Chính Viễn thông, 2009.
2. <https://www.c-sharpcorner.com/article/creating-web-api-and-consuming-in-console-application-using-web-client-in-an-asy/>, truy cập cuối cùng 6/6/2019.
3. <https://dotnettutorials.net/course/asp-net-web-api/>, truy cập cuối cùng 6/6/2019.
4. <https://intellipaat.com/tutorial/kafka-tutorials/application-programming-interface/>, truy cập cuối cùng ngày 6/6/2019.
5. <https://www.tutorialsteacher.com/webapi/web-api-tutorials>, truy cập cuối cùng ngày 28/5/2019.
6. <https://viblo.asia/p/phan-tich-thiet-ke-he-thong-thong-tin-su-dung-bieu-do-uml-phan-1-PjxMe6yNG4YL>, truy cập cuối cùng ngày 28/5/2019.
7. <https://hoclaptrinh.vn/tutorial/hoc-mongodb/tai-lieu-mongodb-tham-khao>, truy cập cuối cùng ngày 1/6/2019.
8. Dương Quang Thiện*, tập 1 – C# căn bản*, 2005.
9. Dương Quang Thiện, *tập 2 – C# và .NET Framework*, 2005.
10. Nguyễn Ngọc Bình Phương và Thái Thanh Phong, *Các giải pháp lập trình C#*
11. <https://aws.amazon.com/vi/nosql/>, truy cập cuối cùng ngày 6/6/2019.
12. <https://quantrimang.com/co-so-du-lieu-phi-quan-he-nosql-160708>, truy cập cuối cùng ngày 6/6/2019.
13. <https://freetuts.net/tong-quan-ve-mongodb-203.html>, truy cập cuối cùng ngày 6/6/2019.