**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**KHOA CÔNG NGHỆ THÔNG TIN**

Logo 
 
Description automatically generated ---------

**BÁO CÁO BÀI TẬP LỚN**

**PHÂN THÍCH THIẾT KẾ HƯỚNG ĐỐI TƯỢNG**

***THIẾT KẾ ĐẶC TẢ YÊU CẦU VÀ MÔ HÌNH HÓA CHO ỨNG DỤNG GHI CHÚ iNote (NOTE-TAKING APP)***

**Giảng viên hướng dẫn: TS. Nguyễn Hiếu Cường**

**Nhóm: 01**

**Lớp: Công nghệ thông tin 2**

**Khóa: 62**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sinh viên thực hiện** | | **:** | **Vũ Nguyễn Trường Giang** | |
| **Mã sinh viên** | | **:** | **211241078** | |
| **Sinh viên thưc hiện** | | **:** | **Trịnh Viết Hân (NT)** | |
| **Mã sinh viên** | | **:** | **211242320** | |
| **Sinh viên thực hiện** | | **:** | **Nguyễn Tiến Toàn** | |
| **Mã sinh viên** | | **:** | **211200830** | |
| **Sinh viên thực hiện** | | **:** | **Nguyễn Phương Anh** | |
| **Mã sinh viên** | | **:** | **211211934** | |
|  | |  |  | |
|  | **Hà Nội 2024** | | |  |

**PHẦN I : XÁC ĐỊNH YÊU CẦU**

**1. Yêu cầu hệ của hệ thống**

**1.1. Yêu cầu tổng quát của ứng dụng iNote**

* Ứng dụng iNote là một ứng dụng ghi chú được xây dựng cho các thiết bị di động, ứng dụng hoạt động song song trên hệ điều hành iOS và Android. Mục tiêu cơ bản là người dùng có thể ghi chú và ghi chép lại các nội dung cần thiết. Ngoài ra, ứng dụng còn hỗ trợ rất nhiều công nghệ để thân thiện với người dùng. Không đơn thuẩn chỉ là ứng dụng ghi chú nhạt nhẽo, ứng dụng iNote được bổ sung các chức năng như thêm ảnh, vẽ hoặc viết bằng tay với apple pen, bảo vệ nội dung nhạy cảm, đồng bộ dữ liệu đến Dropbox, iCloud hoặc Google Drive.

**1.2. Yêu cầu chức năng**

* Người dùng ứng dụng iNote có thể tạo mới một note với dạng nhập chữ đánh máy văn bản, khi tạo người dùng có thể xóa và chỉnh sửa nội dung tùy thích.
* Ngoài hỗ trợ thao tác với văn bản dạng đánh máy thì ứng dụng còn hỗ trợ các tính năng như thêm ảnh vào trong note và tính năng note bằng tay (hand-drawn), điều này yêu cầu người dùng phải có một chiếc Apple Pencil với giá rất chát.
* Các nội dung ghi chú nhạy cảm như mật khẩu ngân hàng, giấy tờ, … có thể được bảo mật khỏi các hacker. Ứng dụng sẽ sử dụng mật khẩu để bảo mật những nội dung như thế này.
* Ứng dụng iNote không chỉ lưu trạng thái của máy một cách độc lập trên máy, ứng dụng có khả năng cho phép đồng bộ dữ liệu mà người dùng đã note tới các máy chủ. Các nền tảng được hỗ trợ bao gồm: Dropbox, iCloud và Google Drive.

**1.3. Yêu cầu phi chức năng**

* Ứng dụng iNote sẽ được triển khai trên hệ điều hành iOS và iPadOS trước tiên. Ứng dụng sẽ hỗ trợ cho các iPhone và iPad có hệ điều hành từ iOS và iPadOS 10 trở lên.
* Ứng dụng sẽ có một trang web support riêng và link tới trang web đó sẽ được ghi ở phần mô tả của ứng dụng.
* Ứng dụng phải có khả năng bảo mật thông tin người dùng, bảo mật trong quá trình thanh toán, độ tín cậy phải cao.
* Hiệu suất của ứng dụng phải có khả năng chịu được nhiều yêu cầu và đi kèm với đó là khả năng phản hồi lại các sự kiện một cách nhanh chóng.
* Giao diện của ứng dụng phải dễ nhìn, màu sắc hài hòa cân đối.

**2. Yêu cầu theo góc nhìn phía người dùng (User Story)**

**2.1. Các chủ đề để thảo luận chính**

Dựa trên các yêu cầu chức năng được đặt ra, chúng ta tổng hợp lại 3 chủ đề chính

để phỏng vấn các người dùng, 3 chủ đề đó là:

1. Thao tác với ứng dụng ghi chú
2. Sự riêng tư – bảo vệ dữ liệu người dùng
3. Dữ liệu được toàn vẹn và đồng bộ

**2.2. Câu trả lời phía các người dùng**

* **Chủ đề #1: Thao tác với ứng dụng ghi chú**
  + **Anh Han Solo**: “Với vai trò là một người dùng cơ bản tôi muốn ứng dụng có thể tạo không giới hạn các note và có thể chỉnh sửa các note tôi muốn sửa để tôi thể nhanh chóng ghi lại những cái ý tưởng hay suy nghĩ của tôi”.
  + **Anh P.Anh Antiantiart**: “Tôi là một người theo trường phái nghệ thuật, ngoài việc ghi chép bằng gõ văn bản ra tôi còn muốn ghi chép bằng tay sử dụng cây bút Apple Pen giá 3 triệu của tôi”.
  + **Anh Toan Cruise**: “Tôi muốn ứng dụng có thể đính kèm hay dán được ảnh vào trong để tôi có thể lưu lại kỉ niệm với người yêu cũ”.
* **Chủ đề #2: Sự riêng tư – bảo về dữ liệu người dùng**
  + **Anh John Lang**: “Tôi muốn ứng dụng khi tôi ghi chú thì có thể bảo mật sao cho chỉ tôi có thể truy cập được các nội dung đó”.
  + **Anh John Luong**: “Tôi muốn bảo vệ các nội dung ghi chú nhạy cảm của tôi với mật khẩu”.
* **Chủ đề #3: Dữ liệu được toàn vẹn và đồng bộ**
  + **Chị Uyen Lisa**: “Tôi là một người di chuyển khá là nhiều nên tôi muốn ứng dụng có thể đồng bộ dữ liệu trên các ứng dụng iOS của tôi, ví dụ như tôi take note bẳng iPad thì khi đi học tôi vẫn có thể lấy điện thoại ra để xem các ghi chú đã ghi trên iPad”.
  + **Anh Quang So Hee**: “Ứng dụng nên được đồng bộ dữ liệu trên các nền tảng lưu trữ đám mây như Dropbox, Google Drive và iCloud để tôi có thể backup tất cả dữ liệu”.

**PHẦN II : MÔ HÌNH CA SỬ DỤNG (USE CASE DIAGRAM)**

*Ở phần mô tả ca sử dụng này, việc mô hình hóa sẽ được chia ra theo 3 chủ đề chính dựa trên các yêu cầu từ phía người dùng cũng như yêu cầu của hệ thống.*

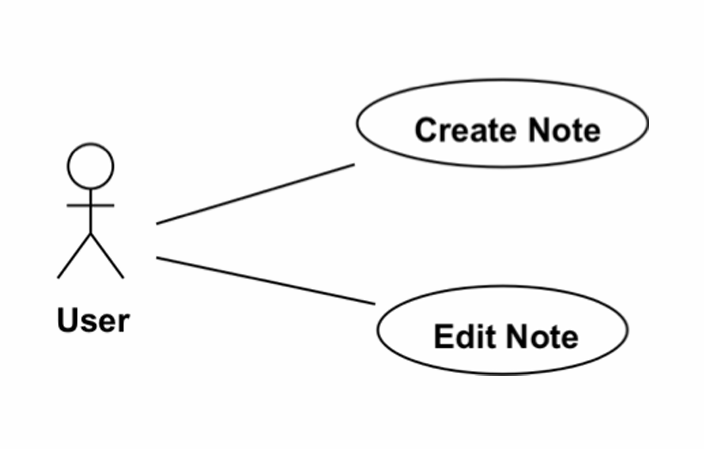
**1. Mô hình ca sử dụng thứ #1: Thao tác với ứng dụng ghi chú iNote.**

Dựa trên các yêu cầu phía người dùng và yêu cầu chức năng, ta nhận thấy tác nhân

hay actor chính của ứng dụng chính là người dùng (User). Với việc có thể tạo mới

một ghi chú hay chỉnh sửa ghi chú thì ta có các use case như “Create Note” hay

“Edit Note”.

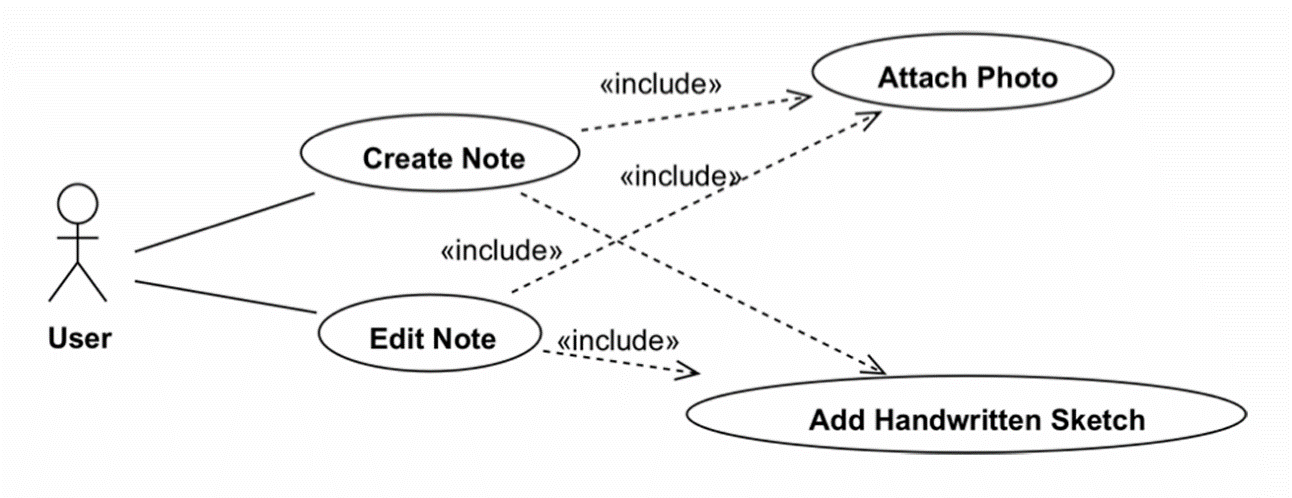


Chúng ta nhận thấy có thêm cả chức năng thêm ảnh (Attach Photo use case) và

chức năng ghi chú bằng bút viết theo kiểu hand-writing (Add Handwritten Sketch

use case). Nhưng chúng ta không thể đính kèm ảnh hay viết vẽ nếu như chúng ta

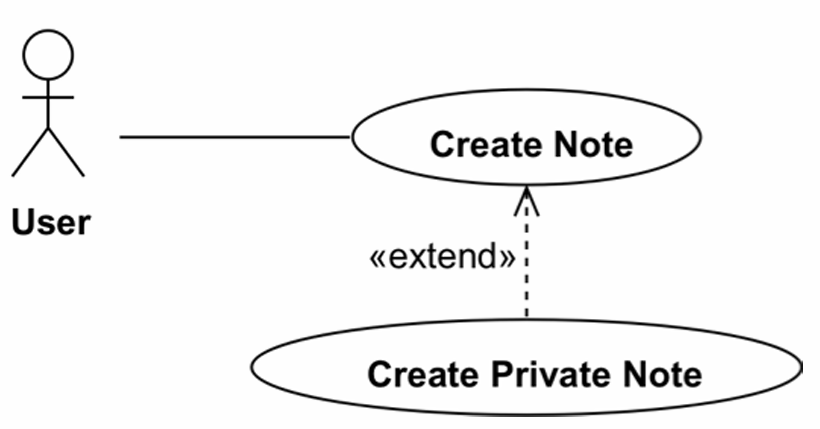
không tạo ghi chú trước đó hay chúng ta đang chỉnh sửa ghi chú. Do đó hai use

 case này sẽ được biểu diễn như một “include”.

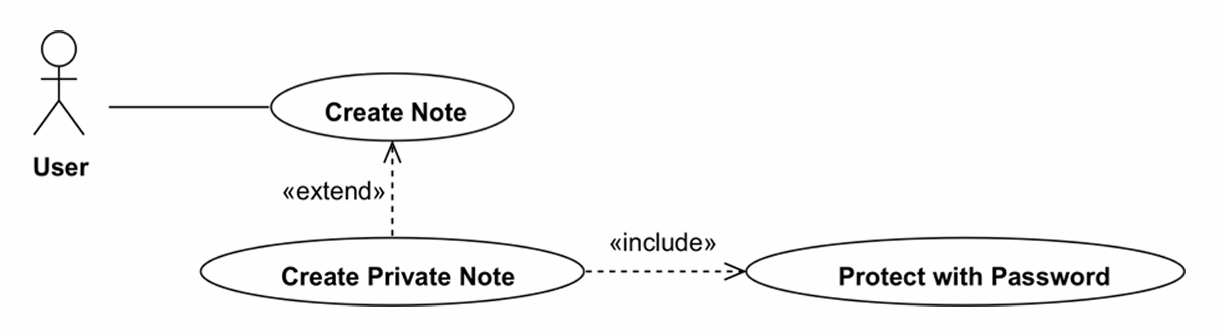
**2. Mô hình ca sử dụng thứ #2: Sự riêng tư, bảo vệ dữ liệu người dùng.**

Ghi chú riêng tư là một use case đặc biệt với ghi chú thông thường. Chúng ta có thể

biểu diễn use case “Create Private Note” là một “extend” của “Create Note”.

****

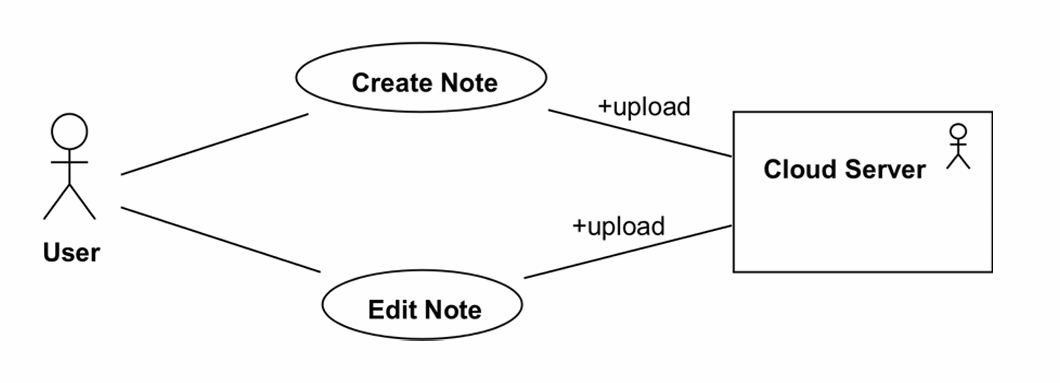
Ngoài ra, chúng ta còn phải bảo vệ những ghi chú nhạy cảm bằng mật khẩu, nên

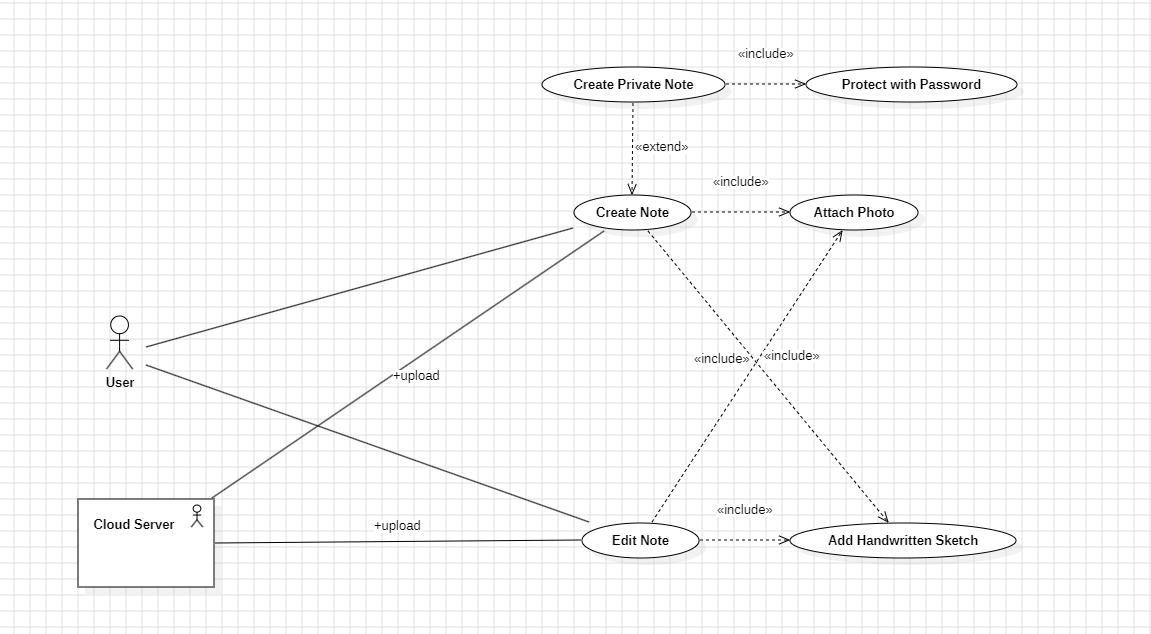
 sẽ có thêm một use case “Protect with Password”. Use case này sẽ “inlcude” với  
 use case “Create Private Note”

**3. Mô hình ca sử dụng thứ #3: Dữ liệu được đồng bộ trên cloud.**

Trong ngữ cảnh của việc đồng bộ với cloud server thì server sẽ là một tác nhân

khác, tác nhân này không phải người dùng hay con người, ta biểu diễn tương tác

 của tác nhân server này với hệ thống app của chúng ta như sau.

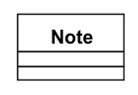
**4. Kết quả**

**PHẦN III : XÁC ĐỊNH MÔ HÌNH LỚP VÀ BIỂU ĐỒ LỚP (CLASS DIAGRAM)**

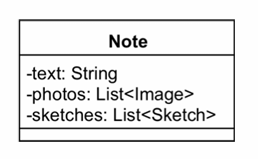
**1. Xác định lớp và biểu đồ lớp cho #1: Thao tác với ứng dụng ghi chú iNote.**

Ứng dụng iNote là một ứng dụng để ghi chú, do đó chúng ta cần một lớp để biểu diễn

ghi chú. Ta có lớp Note



Ngoài ra, lớp Note sẽ có các thuộc tính như text để biểu hiện là chữ cái, hoặc là danh sách các bức ảnh và các chữ viết bằng tay nếu như người dùng có thêm ảnh hoặc viết bằng bút Apple Pen. Ta có thuộc tính text (string), photos (List<Image>) và sketches (List<Sketch>), tất cả đều có thuộc tính private để đảm bảo tính đóng gói dữ liệu.



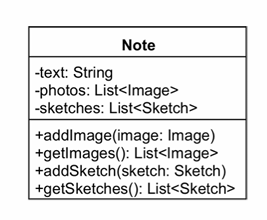
Bên cạnh đó, lớp Note sẽ có các phương thức hay hành vi tương ứng như:

- addImage(image: Image): để đính kèm ảnh.

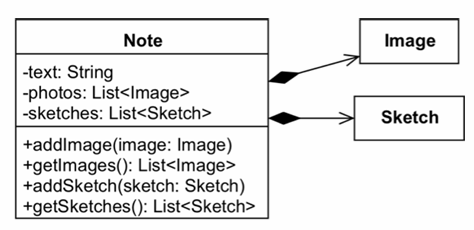
- getImages(): để lấy ra danh sách các ảnh của Note đó.

- addSketch(sketch: Sketch): để thể hiện đang thêm chữ viết bằng bút.

- getSketches(): để lấy ra danh sách các chữ viết bằng bút.



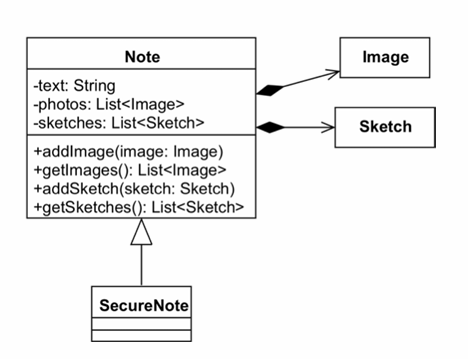
Ta thấy có hai mối quan hệ thành phần (composition) đối với Image, Sketch với Note, vì một khi Note không tồn tại thì các Image hay các Sketch sẽ cũng không tồn tại. Ta có biểu đồ lớp sau



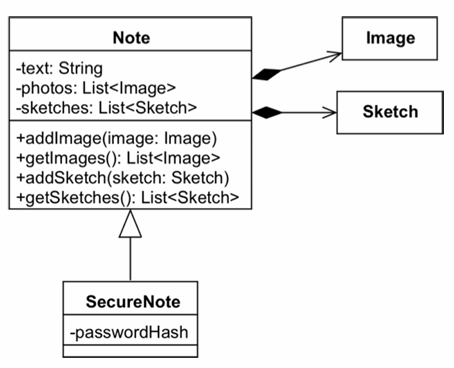
**2. Xác định lớp và biểu đồ lớp cho #2: Sự riêng tư, bảo vệ dữ liệu người dùng.**

Vì vấn đề bảo mật nên sẽ có một lớp Note khác để chứa các ghi chú nhạy cảm cần bảo

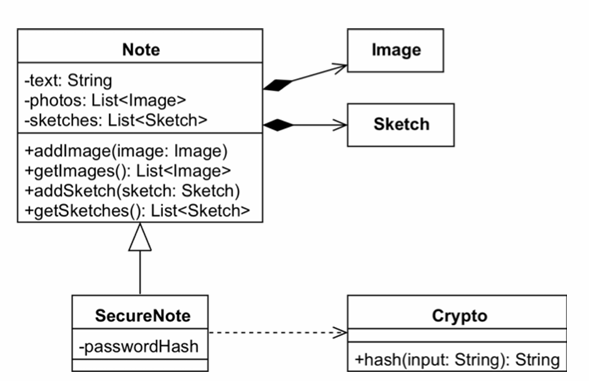
mật nên ta sẽ có thêm một lớp SecureNote kế thừa từ lớp Note là điều hợp lý.



Ngoài các thuộc tính được kế thừa từ lớp Note ra, vì khi xác định yêu cầu chúng ta nhận thấy người dùng Quang So Hee muốn có mật khẩu để mở khóa các nội dung trong ghi chú nên ta sẽ thêm thuộc tính passwordHash cho lớp SecureNote



Bởi vì khi lưu trữ mật khẩu theo dạng text thuần khá là không ổn nên là chúng ta sẽ sử dụng kĩ thuật Hashing để có thể băm (hash) mật khẩu theo một chiều, mật khẩu sẽ không thể tái tạo lại được từ giá trị băm (hash value). Vậy thì chúng ta sẽ cần một lớp Crypto để có thể đảm nhận vai trò như một phụ thuộc (dependency) cho việc nhận vào chuỗi mật khẩu thô và trả lại mật khẩu đã được băm. Vì là mối quan hệ phụ thuộc nên ta có sơ đồ sau.



**3. Xác định lớp và biểu đồ lớp cho #3: Dữ liệu được đồng bộ trên cloud.**

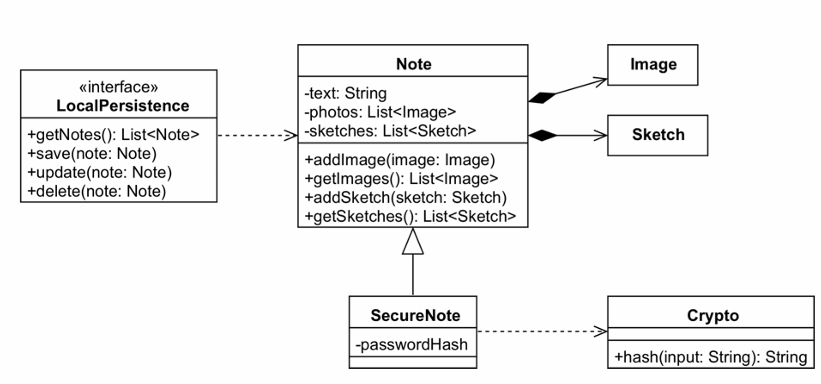
Bây giờ chúng ta cần phải xác định lớp cho việc biểu diễn yêu cầu dữ liệu được đồng bộ trên cloud. Trước hết ta cần xét đến việc lưu dữ liệu cục bộ đã, sẽ có dạng interface để triển khai việc lưu dữ liệu ở cục bộ vì mục tiêu của chúng ta sự tổng quát hóa (generalization) nên sẽ để là interface. Interface đảm nhận cho việc này là LocalPersistence, interface này sẽ có phương thức sau để lớp Note có thể tiêm phụ thuộc (Dependency Injection) để sử dụng cho việc lưu trữ dữ liệu.

- getNotes(): List<Note>: để lấy danh sách các ghi chú từ dữ liệu cục bộ .

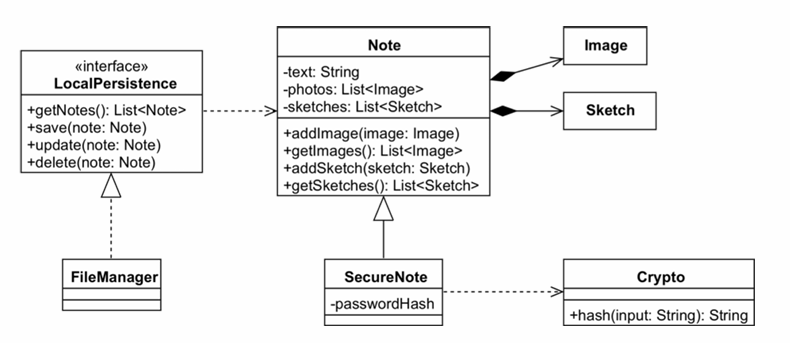
- save(note: Note): để lưu ghi chú trên cục bộ.

- update(note: Note): để cập nhật ghi chú trên cục bộ.

- delete(note: Note): để xóa ghi chú khỏi dữ liệu cục bộ.

Mối quan hệ giữa lớp Note và interface LocalPersistence này là mối quan hệ phụ thuộc, ta có mô hình lớp sau

Ta có thêm một lớp FileManager là một triển khai của interface LocalPersistence để đảm nhận việc triển khai cụ thể lưu trữ các file sẽ như thế nào, lớp Note có thể sử dụng nhiều triển khai của interface LocalPersistence cho việc lưu file cục bộ trên các hệ thống khác nhau.



Sau khi xử lý xong thiết kế các lớp cho việc lưu trữ cục bộ giờ sẽ là phần lưu trữ trên đám mây (cloud server). Tương tự như phần lưu trữ cục bộ chúng ta sẽ tạo ra một interface có tên là NetworkController để có thể quản lý các thao tác lữu trữ đám mây, các thao tác đó sẽ khác với lớp LocalPersistence, các phương thức đại đại diện cho các thao tác đó như sau

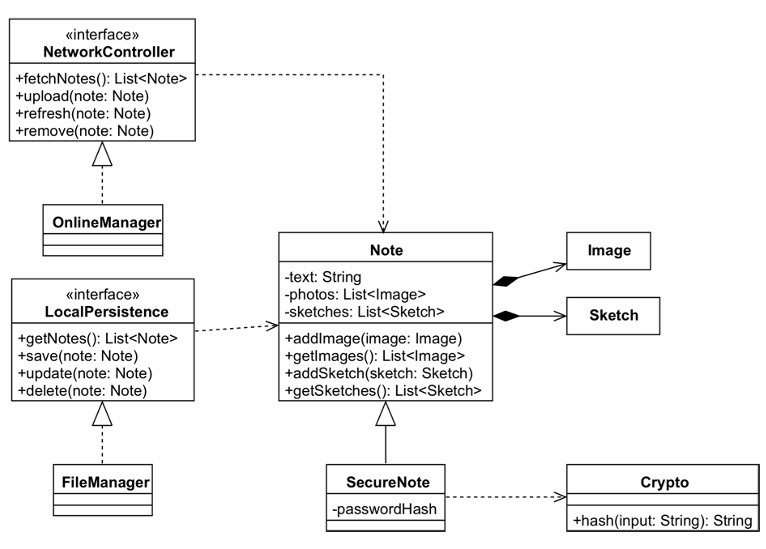
- fetchNotes(): List<Note>: đồng bộ các ghi chú từ phía server vào ứng dụng.

- upload(note: Note): đăng tải các ghi chú từ ứng dụng lên server.

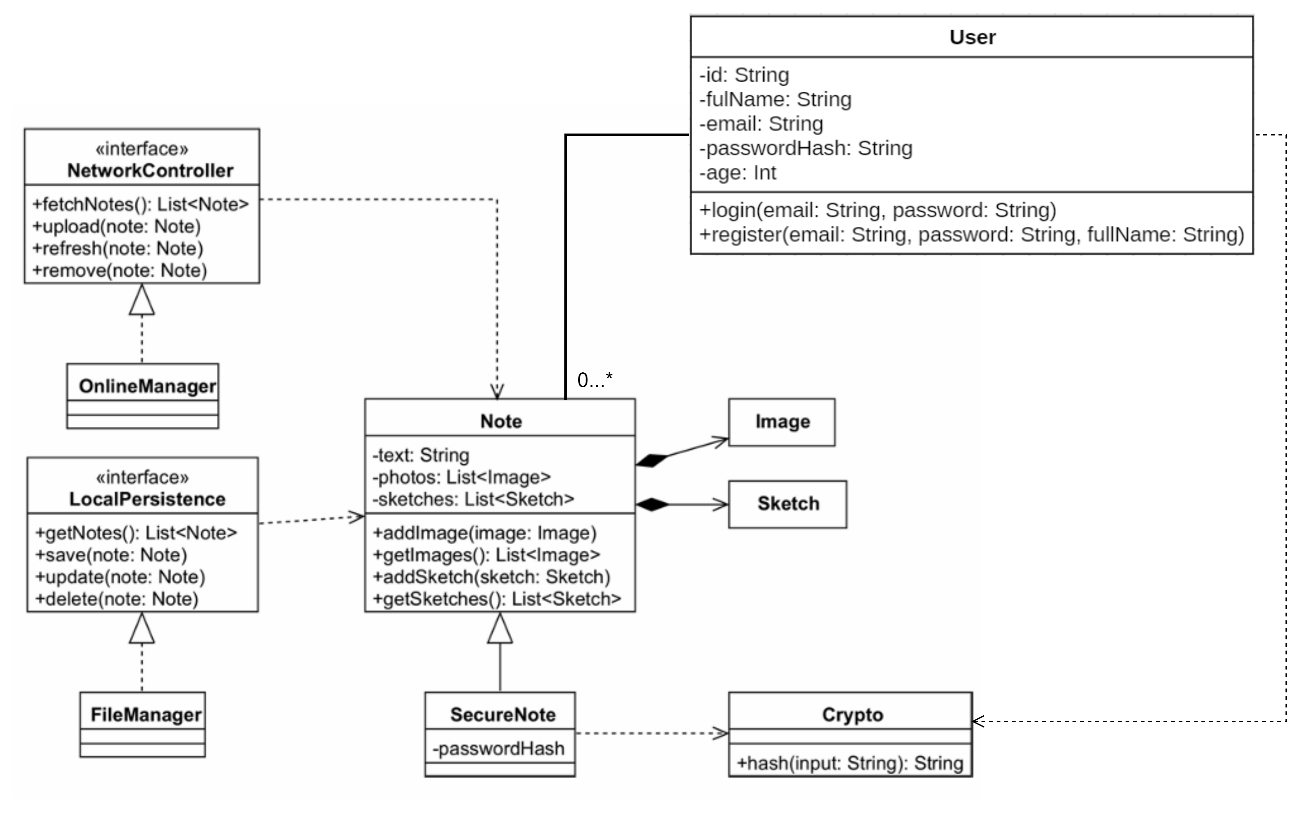
- refresh(note: Note): cập nhật các ghi chú đã được đăng tải trên server.

- remove(note: Note): xóa một ghi chú khỏi server.

Đối với yêu cầu phi chức năng sẽ phải có cơ chế bất đồng bộ nào đó để cho việc đồng bộ (sync) có được hiệu suất cao. Ngoài ra, tương tự như lưu trữ cục bộ chúng ta cần có một triển khai của interface NetworkController để đảm nhận việc triển khai các câu lệnh cho việc đồng bộ dữ liệu hay sửa xóa dữ liệu trên các nền tảng đám mây. Nhận thấy mối quan hệ giữa lớp Note và interface NetworkController là mối quan hệ phụ thuộc (DI) nên ta có sơ đồ hoàn chỉnh sau



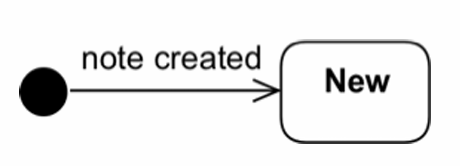
**4. Kết quả.**

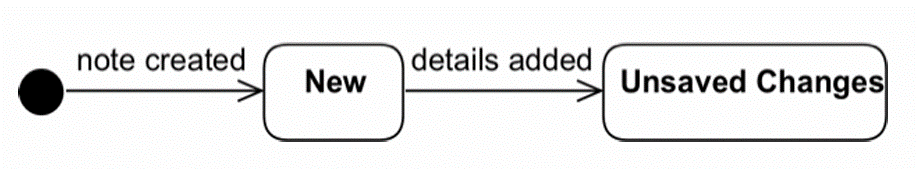
Ta thấy rằng ứng dụng của chúng ta chưa có phần cho người dùng có thể đăng ký, đăng nhập nên ta sẽ tạo thêm một lớp User mới để có thể quản lý người dùng, người dùng được lưu trữ trên firebase.

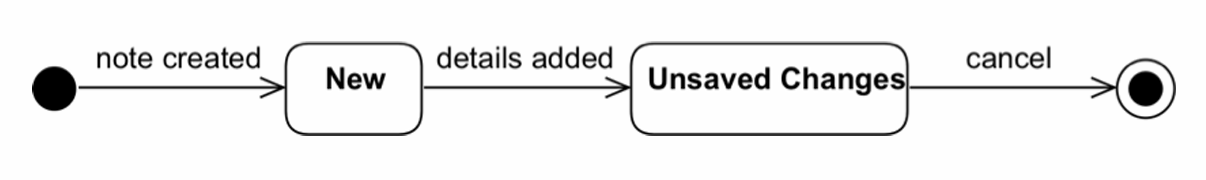
**PHẦN IV : XÁC ĐỊNH BIỂU ĐỒ HÀNH ĐỘNG (ACTIVITY DIAGRAM)**

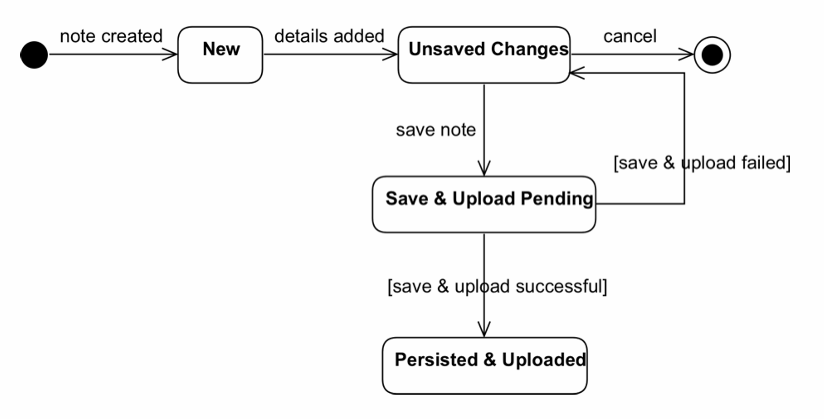
**1. Xác định biểu đồ hành động**

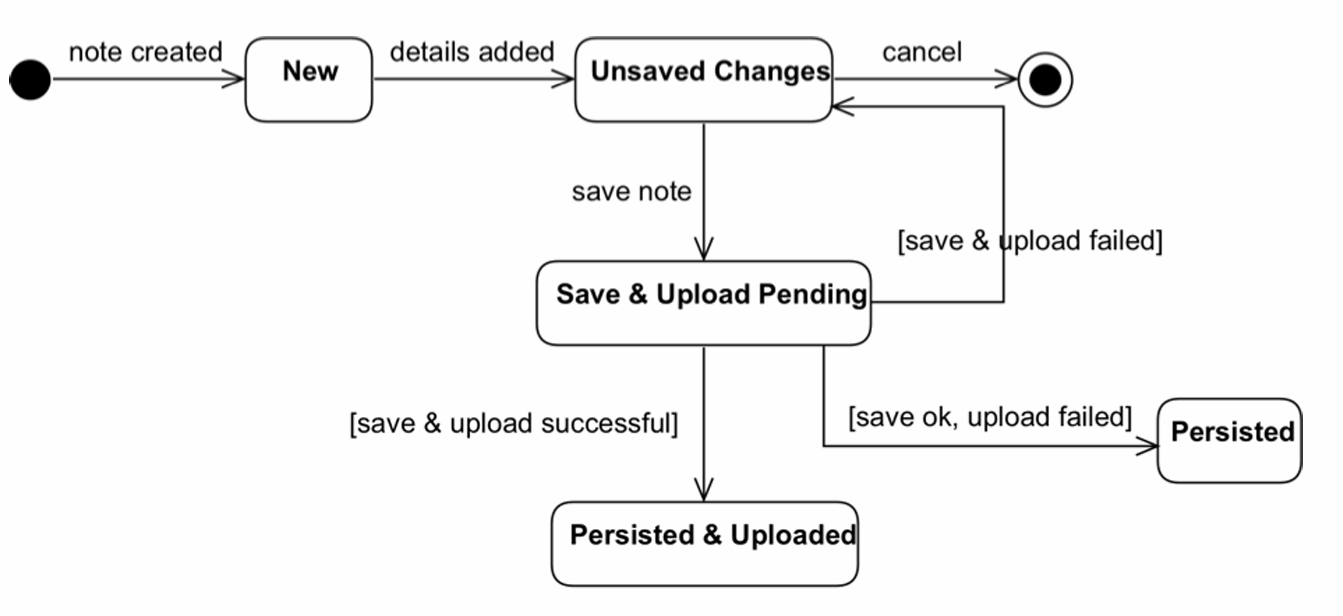
Khi người dùng tạo ra một ghi chú, ghi chú này sẽ ở trạng thái “New”. Đây là sự kiện tạo mới một ghi chú.

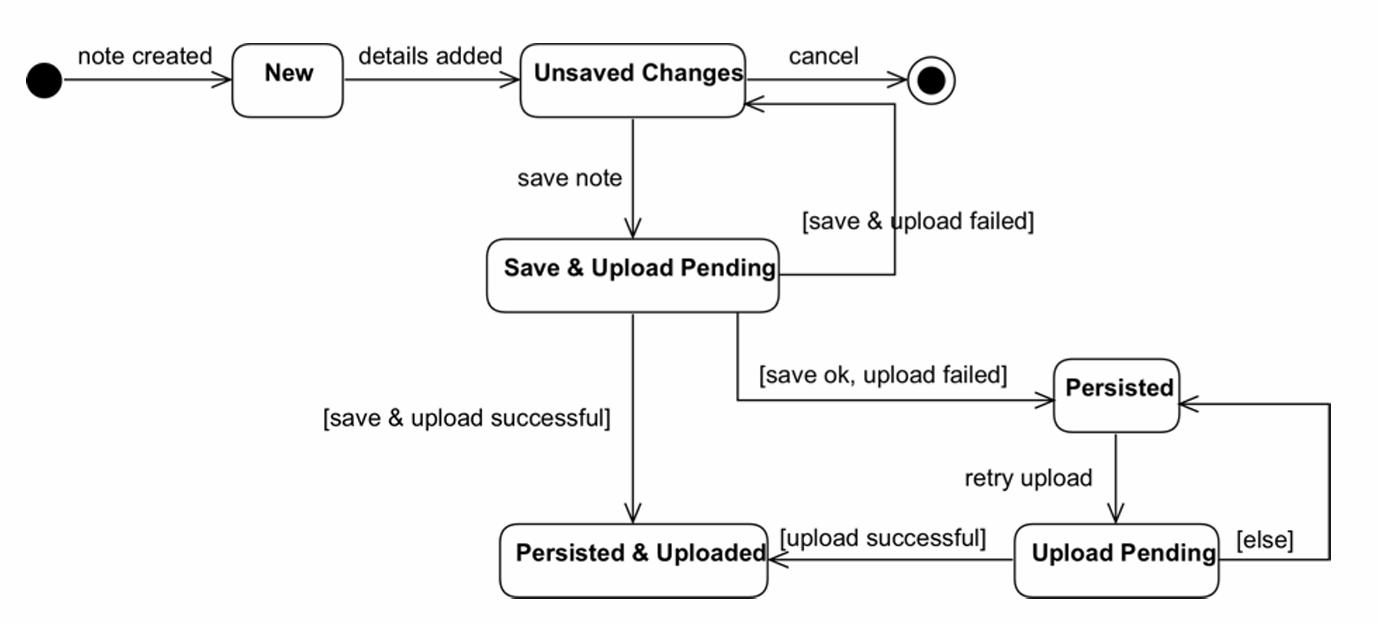


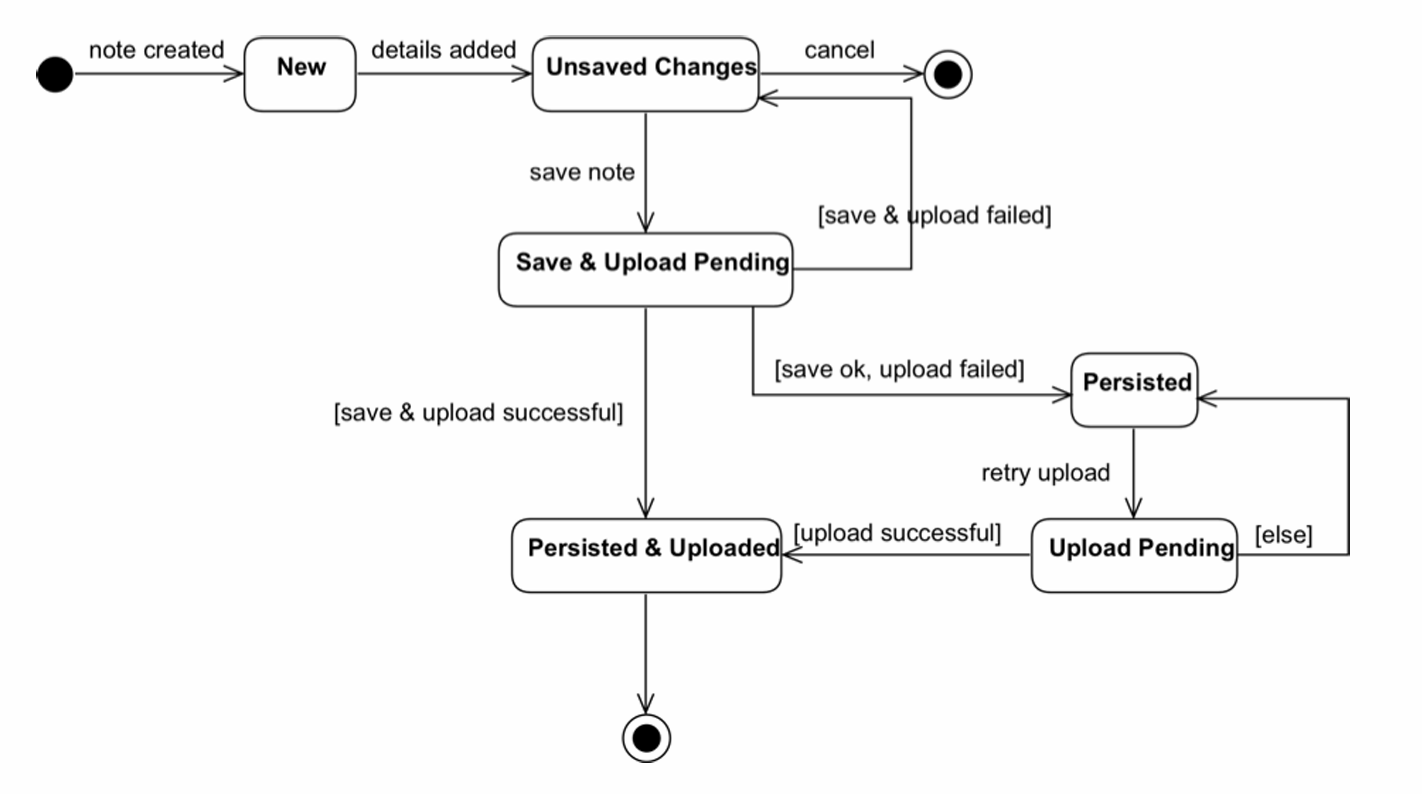
Khi ghi chú mới đã được tạo thì người dùng cần ghi chú, tạo văn bản, điền vào đó các nội dung người dùng muốn ghi chú. Sau quá trình này, ghi chú sẽ có trạng tháng là “Unsaved Changes” tức là những nội dung người dùng ghi chú chưa được lưu.

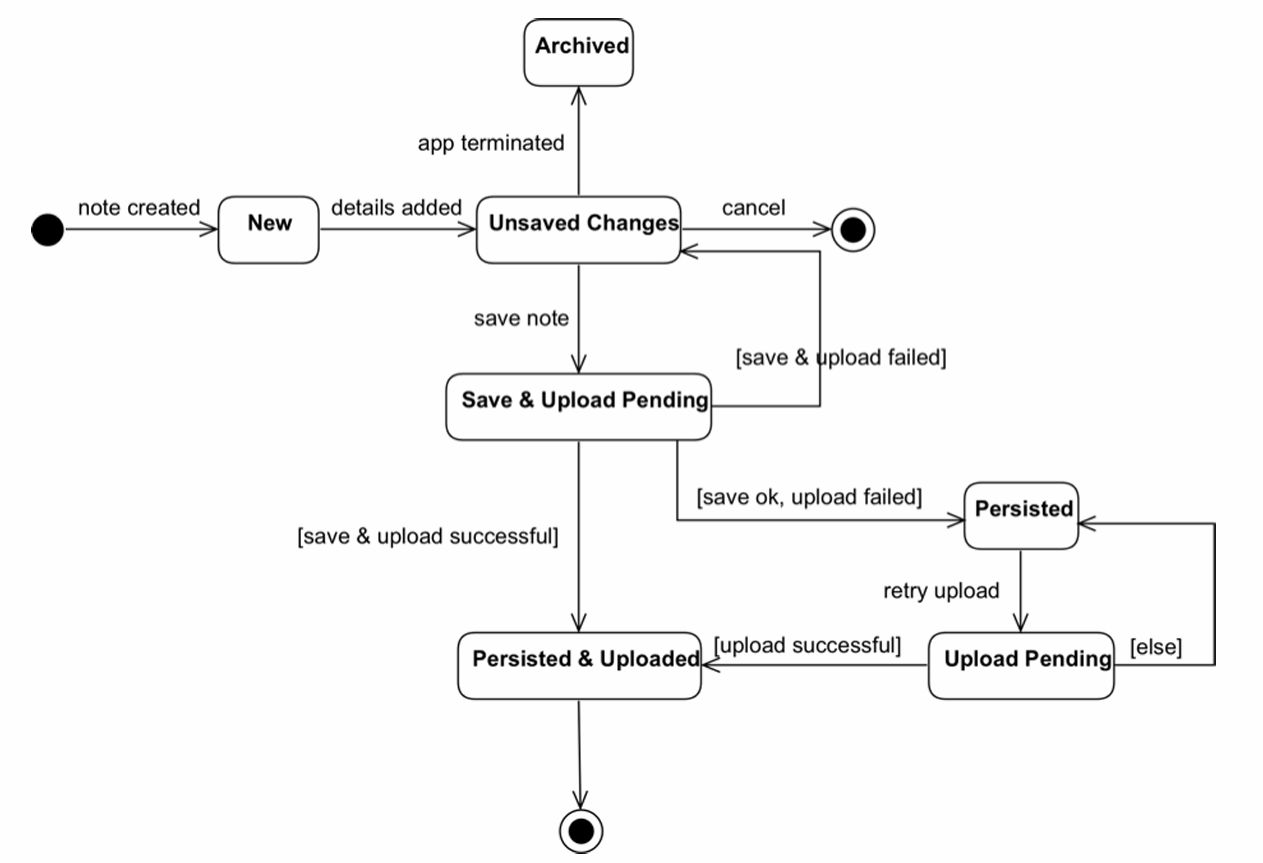
Bây giờ có hai lựa chọn là người dùng có thể lưu ghi chú lại hoặc hủy việc lưu, nếu như người dùng hủy việc lưu ghi chú lại thì tức là biểu đồ hành động của chúng ta đã đi đến trạng thái kết thúc.

Trong trường hợp người dùng lưu ghi chú lại chứ không hủy thì tức là dữ liệu sẽ được lưu trữ trong bộ nhớ cục bộ và bộ nhớ trong đám mây. Dù vậy thì hai hành động này có thể xảy ra lỗi và có khả năng không thành công. Nếu như không thành công thì trạng thái sẽ quay trở lại là “Unsaved Changes”. Ngược lại, nếu thành công thì trạng thái sẽ là “Persisted & Uploaded” tức là đã được lưu xuống cục bộ và đăng tải thành công lên dữ liệu đám mây.

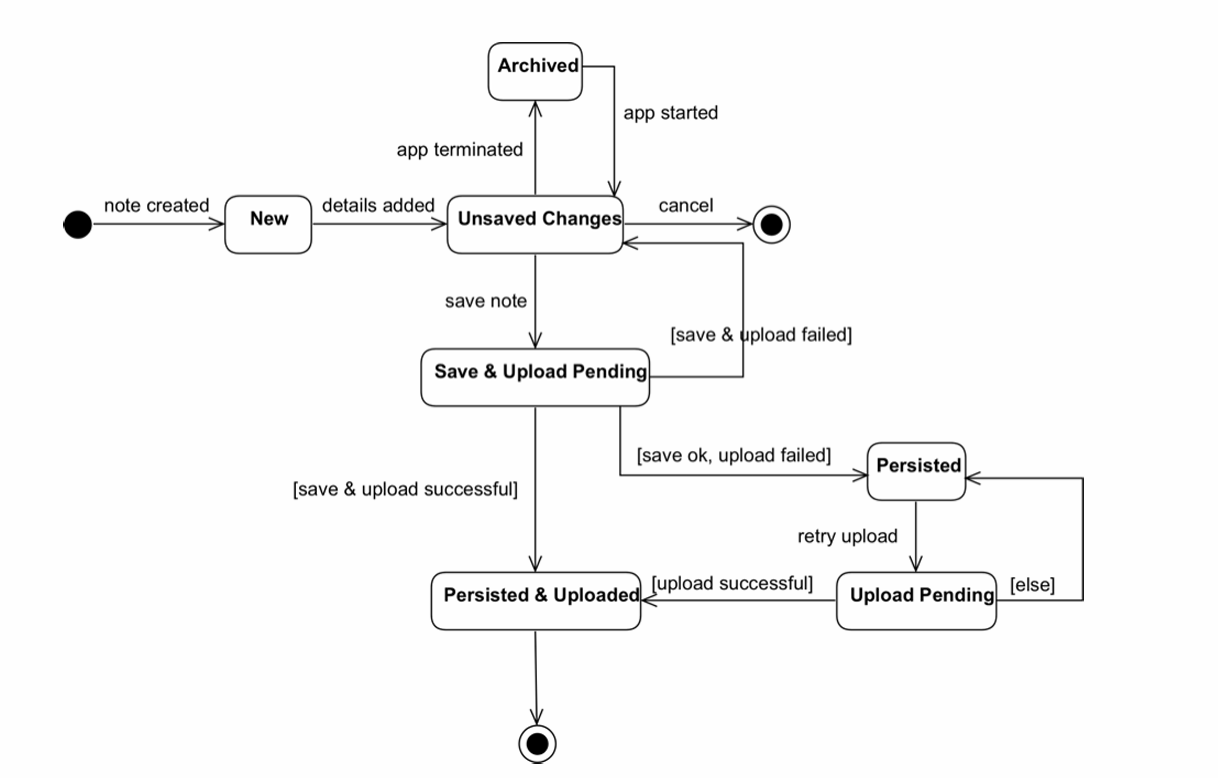
Chúng ta đều đã biết là lưu trữ trong bộ nhớ cục bộ lúc nào cũng có tỷ lệ thành công cao hơn rất nhiều so với là đăng tải hay upload dữ liệu lên đám mây. Vậy nên là đăng tải hay đồng bộ dữ liệu lên trên hệ thống đám mây có thể xảy ra lỗi khá thường xuyên vì một vài lý do nào đó. Lúc này chúng ta sẽ có một trạng thái “Persisted” để chỉ ra rằng là đã lưu trữ cục bộ thành công nhưng mà lưu trữ hay đăng tải trên đám mây thất bại.

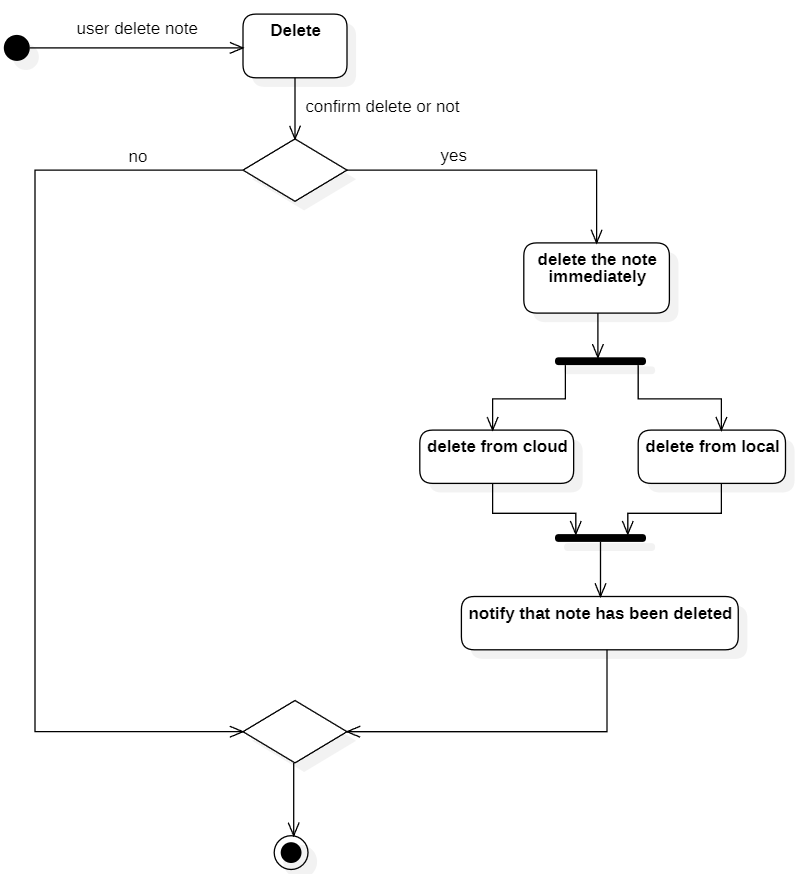
Trong trường hợp lưu trữ cục bộ thành công mà đăng tải lên đám mây thất bại như trên thì lúc này người dùng hoàn toàn có thể thử lại việc đăng tải lên đám mây, lúc này sẽ có thêm một trạng thái mái là “Upload Pending”. Nếu như sau khi thử lại mà vẫn thất bại thì sẽ lặp đi lặp lại trạng thái “Persisted” và sau đó thử lại đến khi nào đăng tải lên đám mây thành công thì thôi. Quá trình này ứng dụng iNote có thể diễn ra một cách tự động, chỉ yêu cầu người dùng xác nhận lỗi và thử lại 1 lần duy nhất, những lần sau mà lỗi như thế thì ứng dụng sẽ cố gắng đăng tải ghi chú lên trên đám mây khi có mạng.

Sau khi ghi chú của chúng ta được tạo ra thành công, được lưu trữ thành công trên cả hai nền tảng thì cuối cùng là trạng thái kết thúc của quá trình.

Ngoài ra, chúng ta còn xét đến trường hợp ghi chú đang ở trạng thái “Unsaved Changes” mà lúc đó người dùng chưa lưu mà lại tắt ứng dụng iNote đi thì lúc này biểu đồ sẽ có thêm một trạng thái mới là trạng thái tạm dừng lưu trữ hay “Archived”. Ứng dụng sẽ chuyển ngay sang trạng thái này nếu như nó bị tắt lúc đang ở trạng thái “Unsaved Changes”

Khi người dùng mở lại ứng dụng iNote vào lần kế tiếp, ghi chú sẽ kích hoạt một sự kiện là “App Started” để từ trạng thái “Archived” quay trở lại trạng thái “Unsaved Changed”, điều này khiến người dùng lần sau muốn chỉnh sửa một ghi chú đã có sẵn cũng sẽ vào biểu đồ hành động này.



Trong trường hợp người dùng muốn xóa một ghi chú thì sẽ phải xác nhận xem có thực sự muốn xóa không. Nếu có thì xóa cả trên cục bộ lẫn đám mây sau đó thông báo còn không thì lại trở về bình thường.