

HƯỚNG DẪN

Kỹ thuật tích hợp ứng dụng vào ứng dụng di động đô thị thông minh Hue-S (Kèm theo Quyết định số: 4101/QĐ-UBND ngày 31 tháng 12 năm 2025 của Ủy ban nhân dân thành phố Huế)

I. TỔNG QUAN

1. Phạm vi đối tượng áp dụng

Tài liệu này nhằm hướng dẫn các cơ quan, đơn vị phát triển các ứng dụng sẽ tích hợp lên nền tảng ứng dụng đô thị thông minh Hue-S.

2. Nền tảng hỗ trợ tích hợp

Hue-S được xây dựng theo hướng siêu ứng dụng đa dịch vụ, sử dụng nền tảng kỹ thuật dựa trên:

- React Native – nền tảng phát triển ứng dụng đa nền tảng (Android/iOS) từ một codebase chung, hỗ trợ tích hợp module native khi cần.

- CodePush (Microsoft App Center) – cơ chế cập nhật nhanh mã JavaScript, CSS và tài nguyên tĩnh mà không cần chờ duyệt App Store/Google Play.

- GitLab & GitLab CI/CD – hệ thống lưu trữ mã nguồn và tự động hóa build, kiểm thử và triển khai.

- GitLab Runner – tác nhân thực thi pipeline tự động hóa build và đẩy bản cập nhật lên CodePush.

Các nền tảng này bảo đảm việc tích hợp Mini App được linh hoạt, giảm chi phí vận hành và rút ngắn vòng đời cập nhật ứng dụng.

STT	Nền tảng ứng dụng bên thứ 3	Tích hợp được Hue-S	Ghi chú
1.	React Native	Hỗ trợ đầy đủ (native module, JS bundle, CodePush)	Khuyến nghị cho Mini App; tương thích kiến trúc Hue-S
2.	Web	Hỗ trợ tích hợp thông qua WebView trong Hue-S	Chủ yếu dùng cho dịch vụ ít tương tác, giao diện tĩnh
3.	Native Android / iOS	Hỗ trợ thông qua SDK hoặc module tích hợp	Yêu cầu đóng gói module native; khó cập nhật qua CodePush
4.	Backend API độc lập	Tích hợp qua Hue-S API Gateway	Đáp ứng chuẩn bảo mật OAuth2

3. Giải thích từ viết tắt

Trong hướng dẫn này các từ ngữ sau đây được hiểu như sau:

STT	Tên viết tắt	Mô tả
1.	Hue-S	Ứng dụng dịch vụ đô thị thông minh Thành phố Huế trên các nền tảng di động
2.	CI	Continuous Integration (Tích hợp liên tục)
3.	CD	Continuous Delivery/Deployment (Triển khai liên tục)
4.	Mini App	Ứng dụng con tích hợp vào siêu ứng dụng
5.	CodePush	Dịch vụ cập nhật mã React Native tức thời
6.	RN	React Native
7.	Pipeline	Quy trình tự động trong GitLab CI/CD
8.	Repository	Kho mã nguồn
9.	Bundle	Gói mã JavaScript được build từ React Native

II. MÔ HÌNH KẾT NỐI

1. Thành phần dùng chung (Component):

Siêu ứng dụng Hue-S sử dụng các thành phần chung nhằm chuẩn hóa việc tích hợp:

- React Native Core Module: quản lý giao diện, điều hướng, tài nguyên.
- Module tích hợp Mini App: cho phép nạp Mini App theo dạng module độc lập.
- CodePush Client: tiếp nhận bản cập nhật ngay khi có thay đổi bundle.
- API Gateway/Backend (tùy mô hình triển khai của từng Mini App).

Các thành phần này đảm bảo Mini App hoạt động thống nhất trong môi trường Hue-S.

2. Sự kiện (Event) để trao đổi dữ liệu

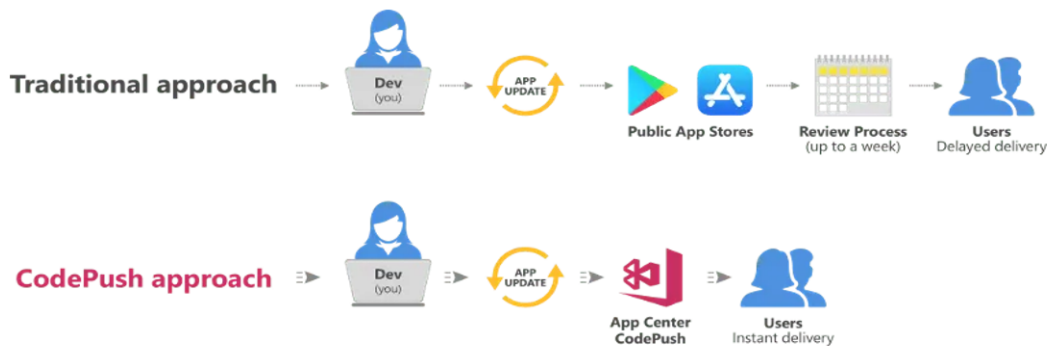
Hue-S sử dụng mô hình event giữa ứng dụng mẹ (container app) và Mini App để:

- Gửi/nhận dữ liệu theo thời gian thực.
- Kích hoạt chức năng, điều hướng hoặc đồng bộ thông tin người dùng.
- Đồng bộ trạng thái Mini App khi có cập nhật từ CodePush.

Một số sự kiện tiêu biểu:

- AppReady, ModuleLoaded, DataSync, UserSessionUpdate, ...

(Tùy đặc tả kỹ thuật từng Mini App, sự kiện sẽ được chuẩn hóa trong tài liệu tích hợp chi tiết.)



Hình: So sánh cập nhật ứng dụng bằng cách truyền thống và Code Push

3. Tạo các kênh kết nối

Mini App được tích hợp vào nền tảng ứng dụng Hue-S thông qua các kênh kết nối:

- Kết nối nội bộ thông qua module React Native (Mini App được đóng gói thành module và nạp tại runtime).
- Kết nối từ xa qua API/Backend (Mini App chỉ gọi API để lấy dữ liệu).
- Kết nối cập nhật qua CodePush (bundle Mini App được build và tự động đẩy lên server).

Hệ thống đảm bảo tích hợp không phá vỡ cấu trúc ứng dụng chính và cho phép cập nhật linh hoạt.

III. QUY TRÌNH TÍCH HỢP CÁC ỨNG DỤNG VÀO NỀN TẢNG ỨNG DỤNG HUE-S

Quy trình chuẩn gồm 5 bước:

1. Chuẩn bị mã nguồn

- Mini App được phát triển theo chuẩn React Native hoặc WebView tùy trường hợp.
- Cấu trúc mã cần tuân thủ các chuẩn module tích hợp của Hue-S.

2. Đăng ký thông tin ứng dụng

- Đơn vị gửi đề xuất tích hợp, mô tả chức năng, API, yêu cầu quyền truy cập.

3. Tích hợp kỹ thuật

- Thực hiện đóng gói Mini App
- Cấu hình các sự kiện trao đổi dữ liệu, truy cập API, điều hướng.

4. Tự động hóa build và kiểm thử (CI/CD)

Khi lập trình viên commit lên GitLab, GitLab Runner sẽ:

- Cài đặt thư viện
- Build bundle React Native
- Chạy kiểm thử
- Đẩy bundle lên CodePush (Staging/Production)

5. Công bố và vận hành

- Mini App được nạp vào siêu ứng dụng thông qua CodePush hoặc bản native mới.
- Theo dõi, kiểm tra
- Vận hành, cập nhật định kỳ.

IV. ĐẶC TẢ CÁC THÀNH PHẦN KỸ THUẬT TÍCH HỢP

1. React Native

- Hỗ trợ đa nền tảng, hiệu năng gần native.
- Dễ tích hợp module native khi cần.
- Codebase chung cho Android và IOS giúp giảm chi phí.

2. CodePush

- Cho phép cập nhật nhanh mà không cần chờ duyệt store.
- Quản lý phiên bản theo môi trường và nhóm người dùng.
- Chỉ thay đổi mã JavaScript và tài nguyên tĩnh.
- Tuân thủ chính sách của Apple/Google về không thay đổi hành vi ứng dụng gốc.

3. GitLab – Repository & CI/CD

- Lưu trữ mã nguồn, quản lý nhánh, version, merge request.
- Thực thi pipeline kiểm thử, build, deploy.
- Tự động hóa toàn bộ quy trình phát hành ứng dụng.

4. GitLab Runner

- Chạy job build bundle React Native.
- Kiểm thử và đẩy bản bundle lên CodePush.
- Kích hoạt pipeline mỗi khi có commit.

V. YÊU CẦU GIAO DIỆN ỨNG DỤNG TÍCH HỢP

Các Mini App tích hợp vào nền tảng Hue-S phải tuân thủ các yêu cầu sau:

1. Đồng bộ giao diện

Không phá vỡ layout và trải nghiệm chung của siêu ứng dụng.

Thiết kế nhất quán về màu sắc, font chữ, phong cách với Hue-S trên các nền tảng khác nhau (Android và IOS) để người dùng có thể dễ dàng sử dụng ứng dụng trên nhiều thiết bị di động. Đối với các giao diện dành cho các thiết bị di động quá khác biệt (đồng hồ thông minh, tivi thông minh...) hay trên máy tính thì cũng cần có những sự tương đồng nhất định với giao diện trên điện thoại.

Không sử dụng quá 3 màu chính (không tính màu đen, trắng và gradient) và cần nhất quán trong việc sử dụng màu. Tuyệt đối không lạm dụng màu sắc khiến cho người dùng khó đọc được nội dung trừ trường hợp màn hình có ít nội dung chữ cần đọc

Không sử dụng nhiều loại font chữ trong ứng dụng, cần sử dụng các loại font, kích thước font và màu chữ sao cho dễ đọc.

Sử dụng các icon đơn giản và hiện đại ví dụ như material icon của google

Bố cục sắp xếp các nội dung ứng dụng cần gọn gàng, dễ nhìn trên nhiều kích thước màn hình và scaling. Phải có sự phân tách rõ ràng giữa các nhóm nội dung trên màn hình.

Hạn chế nhét các điều hướng, chức năng quan trọng hoặc được dùng thường xuyên của ứng dụng ở navigation menu, cố gắng thể hiện các chức năng đó ở màn hình chính.

Không nên thể hiện quá nhiều nội dung trong 1 màn hình nếu không thật sự cần thiết. Nên chia nhỏ các nhóm nội dung ra nhiều màn hình để người dùng dễ đọc và thao tác. Trong trường hợp thiết kế cho máy tính bảng thì 1 màn hình giao diện của máy tính bảng có thể là tổng hợp nhiều màn hình giao diện điện thoại.

Màn hình đầu ứng dụng cần có tên ứng dụng (tên có thể biểu diễn cả dạng text hoặc hình ảnh miễn sao có thể nhận dạng được đây là ứng dụng gì) và nút hamburger để mở navigation menu (cũng có thể mở bằng cách vuốt từ mép trái màn hình).

2. Tối ưu hiệu năng

Thời gian tải Mini App nhanh, không chặn giao diện: Các Mini App tích hợp vào nền tảng Hue-S phải bảo đảm thời gian khởi tạo ngắn, hoạt động ổn định và không gây treo giao diện hoặc chậm phản hồi. Việc tải dữ liệu, khởi tạo chức năng hoặc đồng bộ thông tin phải được thực hiện theo cơ chế bất đồng bộ, hạn chế tối đa việc chiếm dụng tài nguyên hệ thống. Các tác vụ nặng phải được tối ưu hoặc tách nhỏ nhằm tránh ảnh hưởng đến trải nghiệm của người dùng.

Hạn chế sử dụng animation nặng hoặc xử lý lớn trên thiết bị: Ứng dụng con không được sử dụng các animation phức tạp, kỹ thuật xử lý đồ họa hoặc tính toán lớn gây tiêu hao CPU, GPU, đặc biệt đối với các thiết bị cấu hình thấp. Các thuật

toán xử lý dữ liệu dung lượng lớn phải được thực hiện tại phía máy chủ; ứng dụng chỉ tiếp nhận và hiển thị kết quả đã được xử lý. Tài nguyên đa phương tiện như hình ảnh, dữ liệu tải về phải được tối ưu để giảm tải cho thiết bị người dùng.

3. Thân thiện người dùng

Dễ sử dụng, thống nhất cấu trúc điều hướng: Các Mini App phải tuân thủ cấu trúc điều hướng chung của nền tảng Hue-S, bảo đảm thống nhất về cách thức di chuyển giữa các màn hình, cách bố trí các nút chức năng, tiêu đề và các thành phần giao diện khác. Ứng dụng cần được thiết kế theo nguyên tắc đơn giản, dễ tiếp cận, hạn chế tối đa số thao tác của người dùng trong mỗi quy trình nghiệp vụ.

Tự động điều chỉnh giao diện cho Android/iOS: Giao diện của Mini App phải được thiết kế tương thích với các độ phân giải và tỉ lệ màn hình khác nhau trên hai hệ điều hành Android và iOS. Các thành phần giao diện không được sử dụng kích thước cố định; phải áp dụng cơ chế co giãn linh hoạt nhằm bảo đảm hiển thị thống nhất trên nhiều loại thiết bị. Những khác biệt đặc thù của từng hệ điều hành (font, định dạng nút, hành vi điều hướng...) phải được xử lý theo tiêu chuẩn chung của nền tảng Hue-S.

4. Chuẩn hóa luồng hoạt động

Hỗ trợ sự kiện chung để Hue-S điều hướng và đồng bộ dữ liệu: Các Mini App phải hỗ trợ đầy đủ các sự kiện (event) theo chuẩn được quy định bởi nền tảng Hue-S nhằm bảo đảm khả năng trao đổi dữ liệu, đồng bộ trạng thái và điều hướng giữa ứng dụng con và ứng dụng chính. Các sự kiện này bao gồm nhưng không giới hạn: sự kiện phiên người dùng (session), sự kiện điều hướng, sự kiện đồng bộ dữ liệu và các sự kiện vòng đời ứng dụng. Ứng dụng con không được tự ý xây dựng cơ chế truyền thông riêng gây xung đột với hệ thống.

Tương thích với cơ chế cập nhật CodePush: Các Mini App phải bảo đảm khả năng cập nhật linh hoạt thông qua hệ thống CodePush của Hue-S, bao gồm việc đóng gói bundle, cập nhật mã JavaScript và tài nguyên tĩnh mà không yêu cầu phát hành lại ứng dụng trên các kho ứng dụng di động. Ứng dụng con không được tích hợp các thành phần native mới ngoài phạm vi cho phép, trừ trường hợp đã có sự thống nhất với đơn vị vận hành để đảm bảo không vi phạm chính sách của Apple và Google. Mọi thay đổi phải tương thích với chuẩn cập nhật của Hue-S, bảo đảm hoạt động ổn định sau khi triển khai./.