



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

Mobile Computing

1st Lecture

Lecture Overview



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



“모바일컴퓨팅” 수업의 목적은 최근 모바일 서비스 개발의 주요 Framework으로 사용되고 있는 React Native를 이용해 빠르게 화면을 구성하고 Backend 자원을 효과적으로 연동 시킬 수 있는 이해도를 높이기 위한 수업이다. 모바일 Frontend 개발의 부담을 줄이고 Workflow 및 서비스 내용에 집중 할 수 있도록 하여 모바일 서비스의 구조를 이해하고 개발할 수 있는 능력을 배양한다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



평가	평가방법	평가시기(주차)	평가비율(%)	평가내용
	지필시험(중간)	8	30	수업내용 이해도
	지필시험(기말)	15	30	수업내용 이해도
	수행과제-실험,실습,실기	7	10	개념기획, 구현명세 작성 실무개발 능력
	수행과제-개발결과물	14	20	설계/구현 능력
	출석	결석감점점수 1점	10	
총합			100	



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



차시	수업주제	수업내용
1	강의소개	하이브리드 앱 기반 모바일 프로그래밍 소개
2	개발환경 구축	리액트 네이티브 개발환경 구축
3	컴포넌트 이해	JSX, Component, props/state, event
4	분석/설계	서비스 아이디어 기반 스토리보드 작성, 화면설계
5	스타일링	React native style, Styled component
6	클라우드 기반 백엔드 실습 1	클라우드 기반(GCP) Back-end 구축 (VM,WAS,DBMS Deploy)
7	클라우드 기반 백엔드 실습 2	클라우드 기반(GCP) Back-end 실습 (DBMS 테이블 작성 WAS 서비스 실습)
8	중간고사	중간고사
9	어플리케이션 설계/구현	프로젝트/입력 컴포넌트/일정관리 앱 설계/구현
10	Hook	useState/useEffect/useRef/useMemo/커스텀 Hooks 만들기
11	Context API	전역 상태 관리/Context API/useContext
12	네비게이션	리액트 네비게이션/스택 네비게이션/탭 네비게이션
13	채팅 어플리케이션	파이어베이스/앱 아이콘과 로딩 화면/인증 화면/메인 화면
14	배포	프로젝트 빌드/iOS 배포/안드로이드 배포
15	기말고사	기말고사



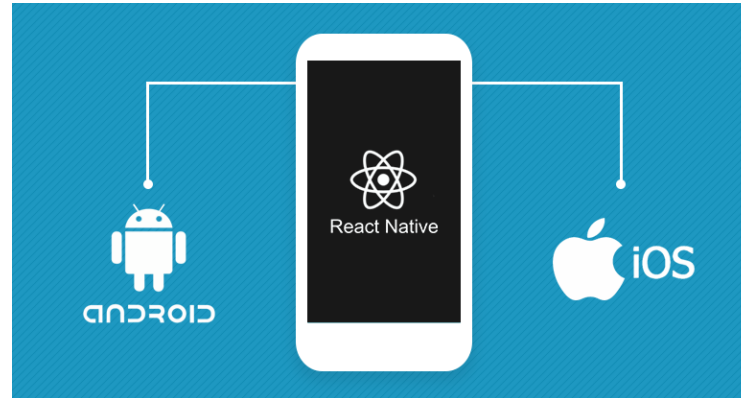
Department of
Media Software

2023 1st semester
Mobile Computing

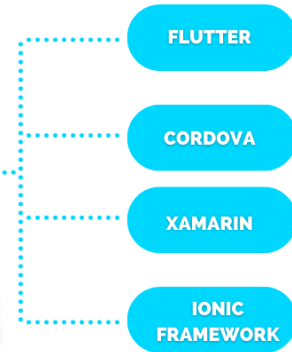
Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



Top 4 React Native ALTERNATIVES



Back4App



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



iOS



JavaScript

Android





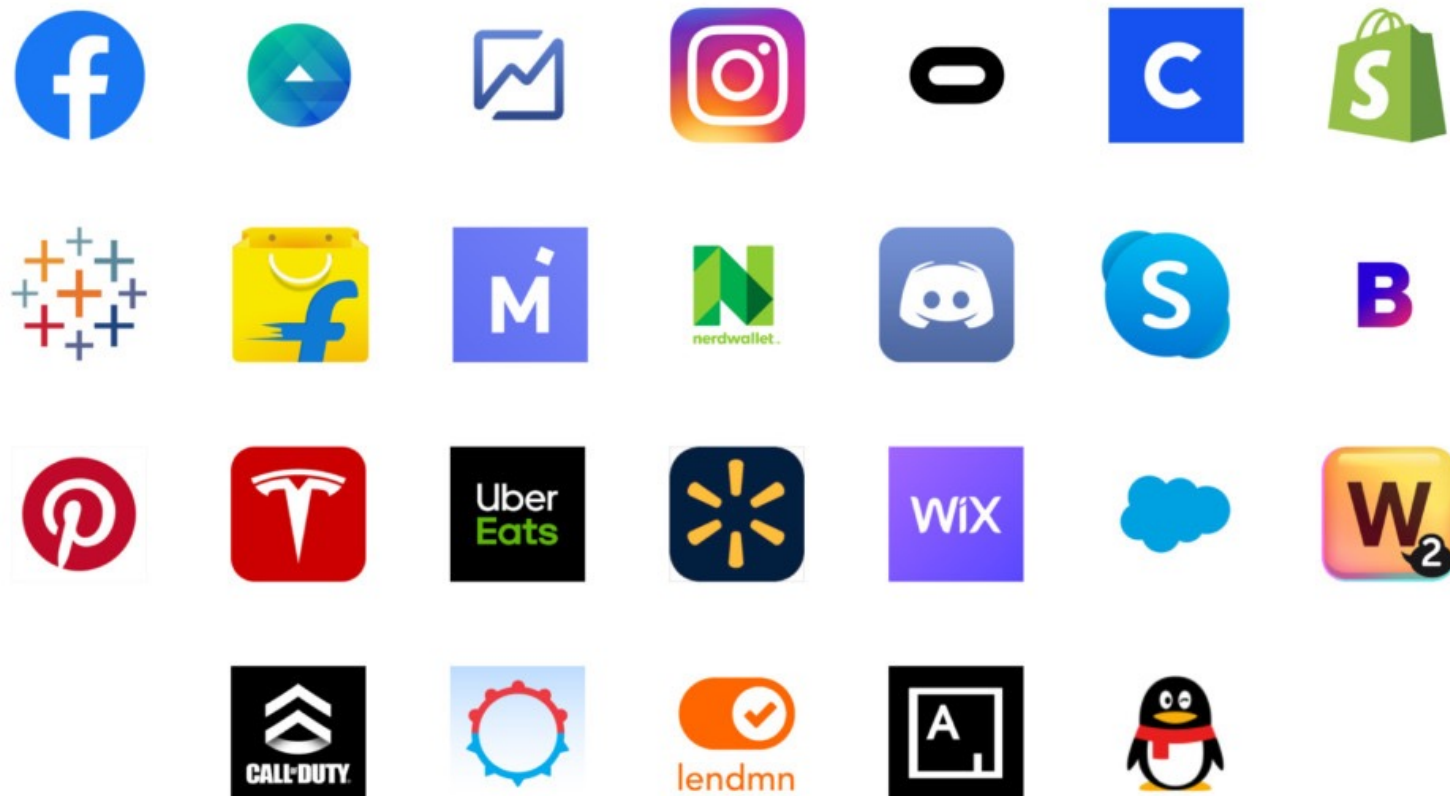
Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



Apps using React Native



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



동작 원리

리액트 네이티브의 동작 원리는 리액트 네이티브가 가상 DOM을 통해 DOM을 조작하지 않는다는 점을 제외하고는 리액트와 실질적으로 동일하다. 직렬화, 비동기, 일괄 처리 브리지를 통해 네이티브 플랫폼과 통신하며 종단 장치에 직접(개발자가 작성한 자바스크립트를 해석하는) 백그라운드 프로세스로 실행된다.

리액트 구성 요소들은 리액트의 선언형 UI 패러다임과 자바스크립트를 통해 기존 네이티브 코드를 감싸고 네이티브 API와 통신한다.

이렇게 함으로써 새 개발자 팀을 위한 네이티브 앱 개발을 가능케 하며 기존 네이티브 팀들이 훨씬 더 빠르게 작업할 수 있게 한다.

리액트 네이티브는 HTML이나 CSS를 사용하지 않는다. 그 대신 자바스크립트 쓰레드로부터의 메시지를 사용하여 네이티브 뷰를 조작한다. 리액트 네이티브는 개발자들이 안드로이드용 자바/코틀린, iOS용 오브젝티브-C/스위프트와 같은 언어로 작성한 네이티브 코드와 인터페이스가 가능하도록 브릿지(Bridge)를 제공하므로 더 많은 유연성을 제공한다.



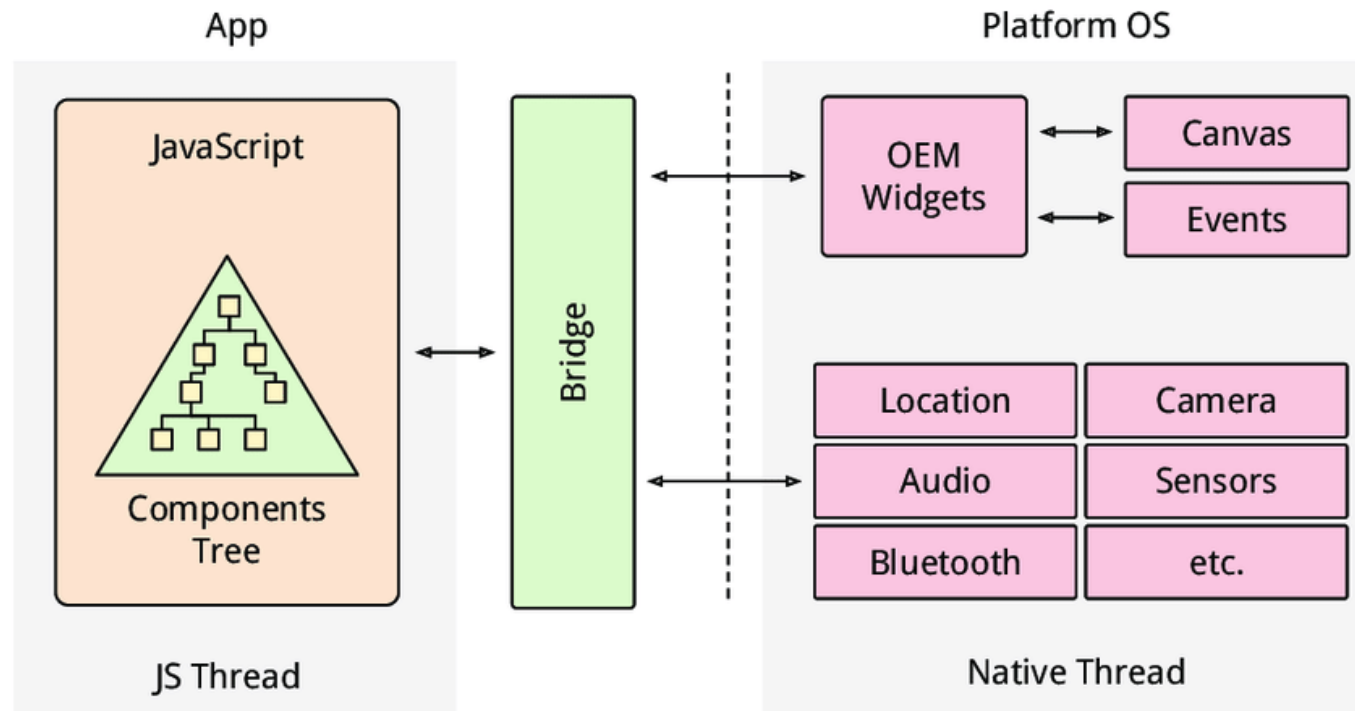
Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing





Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

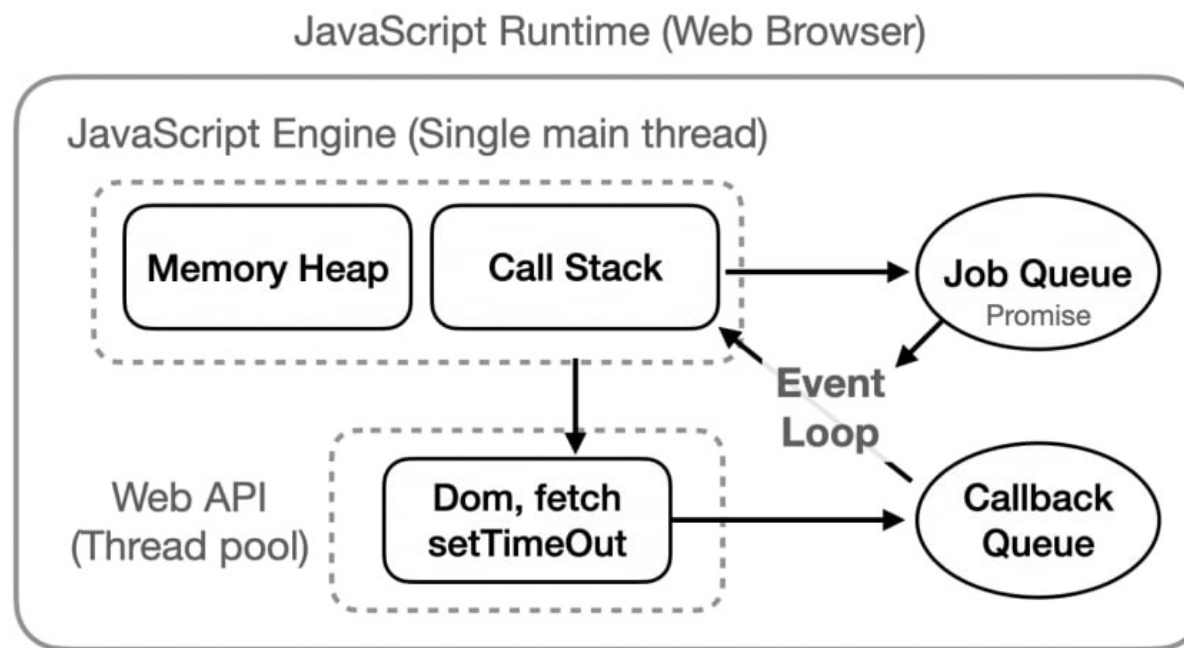
1st Lecture

Mobile Computing



실행 환경

React는 JavaScript 런타임 환경에서 실행됩니다. 웹의 경우 웹 브라우저입니다. 단일 JavaScript 스레드가 있으며 브라우저에 기본적으로 구현된 웹 API를 사용합니다.



특히 비동기 함수를 사용할 때 기본 JS 스레드와 웹 API 간에 콜백이 작동하는 방식을 이해하는 것이 중요합니다. JavaScript 엔진과 네이티브 API 간의 이러한 상호 작용은 React Native가 해당 환경에서 어떻게 작동하는지 이해하는 데에도 중요합니다.

Ref : <https://dev.to/goodpic/understanding-react-native-architecture-22hh>



Department of
Media Software

2023 1st semester
Mobile Computing

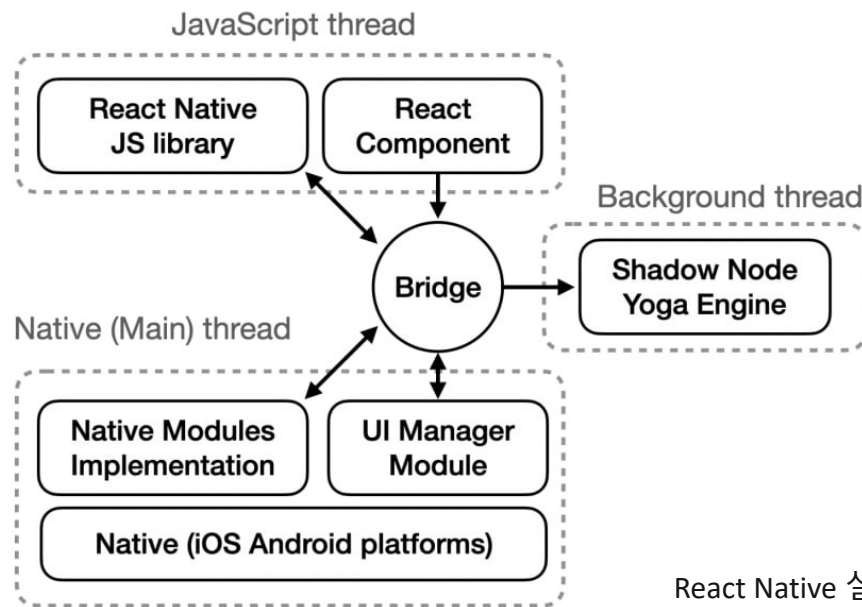
Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



React Native의 스레드



React Native 실행 환경에는 JavaScript 스레드, Native 메인 스레드 및 Shadow Node를 처리하는 백그라운드 스레드의 세 가지 핵심 스레드가 있습니다. 현재 아키텍처에서 이러한 스레드 간의 통신은 "브리지"라는 라이브러리를 통해 발생합니다.

React Native 팀은 주요 아키텍처 업그레이드에 적극적으로 참여하고 있으며 이 기사에서는 변경이 필요한 이유와 방법에 대한 훌륭한 개요를 제공합니다. 자세히 설명하지는 않지만 현재와 미래의 아키텍처에 대한 기본적인 이해는 애플리케이션을 모델링하는 데 도움이 되며 특히 문제를 분리하는 데 도움이 됩니다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



반응 및 반응 네이티브

위에서 볼 수 있듯이 React Native는 React 자체보다 훨씬 더 넓은 영역을 다룹니다. 웹용 React는 핵심 개념인 가상 DOM이 브라우저의 HTML 돔 렌더링을 암시하므로 다소 직관적으로 보입니다. 그러나 사실 Virtual DOM은 HTML DOM(Document Object Model)에 연결되어 있지 않습니다. React의 가상 DOM은 특정 기술이라기보다는 프로그래밍 개념이나 패턴에 가깝습니다.

선언적 UI에 대한 추상화를 제공합니다. UI의 가상 표현은 메모리에 보관되고 외부 UI 라이브러리와 동기화됩니다. 이 프로세스를 조정 이라고 합니다. 여기에서 React Fiber 아키텍처에 대한 좋은 설명을 읽을 수 있습니다.

조정 대 렌더링

DOM은 React가 렌더링할 수 있는 렌더링 환경 중 하나일 뿐이며 다른 주요 대상은 React Native를 통한 기본 iOS 및 Android 보기입니다. (이것이 "가상 DOM"이 약간 잘못된 이름인 이유입니다.)

이렇게 많은 대상을 지원할 수 있는 이유는 React가 조정과 렌더링이 별도의 단계로 설계되었기 때문입니다. 조정자는 트리의 어떤 부분이 변경되었는지 계산하는 작업을 수행합니다. 그런 다음 렌더러는 해당 정보를 사용하여 렌더링된 앱을 실제로 업데이트합니다. 이러한 분리는 React DOM과 React Native가 React 코어에서 제공하는 동일한 조정자를 공유하면서 자체 렌더러를 사용할 수 있음을 의미합니다.

Fiber는 조정자를 다시 구현합니다. 렌더러가 새로운 아키텍처를 지원하고 활용하기 위해 변경해야 하지만 주로 렌더링과 관련이 없습니다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

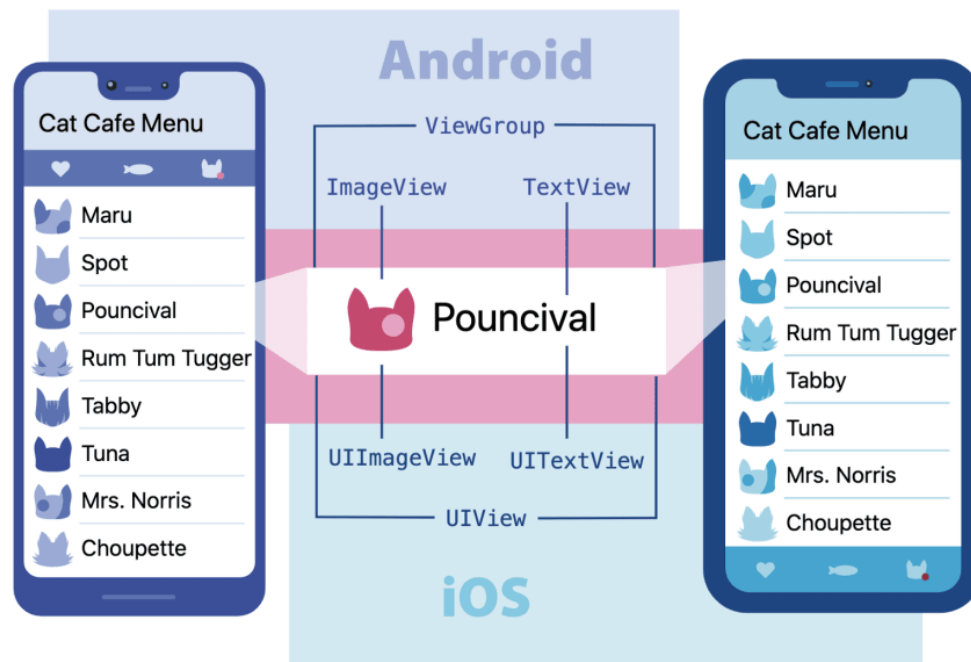
1st Lecture

Mobile Computing



기본 구성 요소 반응 및 탐색 반응 반응

React Native는 iOS 및 Android 플랫폼을 통해 자체 UI 추상화 계층을 제공합니다. React Native 핵심 및 기본 구성 요소는 기본 보기를 호출하므로 Kotlin/Java 또는 Swift/Objective-C 대신 JavaScript로 스마트폰 앱 UI를 작성할 수 있습니다.





Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



핵심 구성 요소 및 기본 구성 요소

기본 구성 요소는 포괄적인 기본 UI 요소를 포함하지만 탭 탐색과 같은 기본 사용자 경험을 시뮬레이션하려면 여전히 많은 코드를 작성해야 합니다. 바로 여기에서 React Navigation이 등장합니다.

React Navigation은 네이티브 코드를 포함하지 않는 순수한 JavaScript 라이브러리입니다. 일반 앱 탐색 패턴을 구현하기 위해 Reanimated, Gesture Handler 및 Screens와 같은 다른 기본 라이브러리에 빌드됩니다.

웹 개발 배경에서 왔을 때 가장 혼란스러운 부분 중 하나인 앱 화면을 구조화하고 탐색하는 방법에 대한 모범 사례를 제공합니다. 제 조언은 자신감이 생길 때까지 기본 탐색 패턴을 고수하고 좋은 개요가 있으면 React 탐색 위에 사용자 지정 탐색기를 구현할 수 있다는 것입니다. 또한 네비게이터와 화면을 전용 디렉토리에 배치하여 다른 구성 요소와 명확하게 구분하는 것을 선호합니다.

React에서 생각하기

UI 구현의 차이에도 불구하고 새로운 애플리케이션을 구축하는 사고 과정은 ["Thinking in React" 방식](#)과 동일하게 유지됩니다.

- 모의로 시작
- UI를 구성 요소 계층 구조로 나누기
- React에서 정적 버전 빌드
- UI 상태의 최소(그러나 완전한) 표현 식별
- 귀하의 주에 거주해야 하는 위치 확인
- 역 데이터 흐름 추가



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



후크 및 기능 구성 요소

React 16.8.0은 2019년에 Hooks를 도입했으며 큰 패러다임 전환이었습니다. React 팀은 Hooks가 모든 클래스 구성 요소 사용 사례를 대체 할 것으로 예상하고 인기 있는 라이브러리는 이미 이 방향으로 마이그레이션했습니다. 예를 들어 React Navigation 5.0 및 React Redux 7.1.0은 Hook API를 도입했습니다.

개념적으로 React 구성 요소는 항상 함수에 더 가깝고 "Thinking in React" 방식은 Hooks를 통해 더욱 간단해졌습니다.

Hooks의 동기는 다음과 같은 이점을 설명합니다.

후크를 사용하면 수명 주기 방법에 따라 분할을 강제하는 대신 관련된 부분(예: 구독 설정 또는 데이터 가져오기)에 따라 하나의 구성 요소를 더 작은 기능으로 분할할 수 있습니다. 더 예측 가능하도록 리듀서를 사용하여 구성 요소의 로컬 상태를 관리하도록 선택할 수도 있습니다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



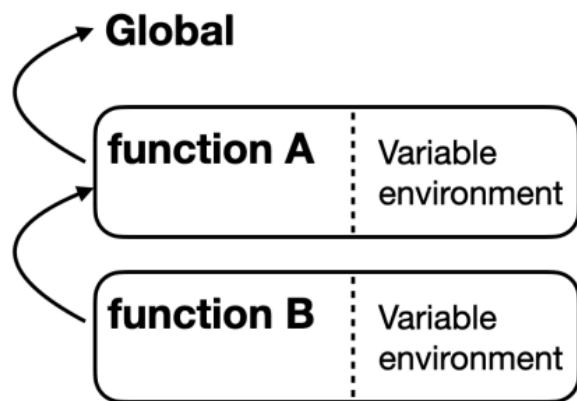
구성 요소 부작용 및 종료

프로토타입 상속을 기반으로 JavaScript 함수는 일급 시민이라고 합니다. 즉, 다음과 같습니다.

- 변수에 할당
- 기능 매개변수로 사용
- 함수에서 반환

이는 React의 Functional Component에도 동일하게 적용됩니다. JavaScript의 Closure도 Hooks를 사용할 때 필수적인 요소입니다.

Scope Chain



Closure

```
function A () {  
  const name = 'closure'  
  function B () {  
    alert(name)  
  }  
}
```

클로저는 주변 상태(어휘적 환경)에 대한 참조와 함께 묶인(포함된) 함수의 조합입니다. 즉, 클로저는 내부 함수에서 외부 함수의 범위에 대한 액세스를 제공합니다. JavaScript에서 클로저는 함수 생성 시 함수가 생성될 때마다 생성됩니다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



컴포넌트 라이프사이클에서 언제 클로저가 생성되는지 이해하고 Hooks 내에서 불안정한 상태 변수 대신 안정적인 함수를 사용하는 것이 매우 중요합니다.

```
function Counter() {
  const [count, setCount] = useState(0);

  useEffect(() => {
    const id = setInterval(() => {
      setCount(c => c + 1); // ✅ This doesn't depend on `count` variable outside
    }, 1000);
    return () => clearInterval(id);
  }, []); // ✅ Our effect doesn't use any variables in the component scope

  return <h1>{count}</h1>;
}
```

참고: 클래스 구성 요소에 있는 `setState` 메서드와 달리 `useState`는 업데이트 개체를 자동으로 병합하지 않습니다. 함수 업데이트 형식을 개체 확산 구문과 결합하여 이 동작을 복제할 수 있습니다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



React는 또한 현재 인증된 사용자, 테마 또는 기본 언어와 같은 React 구성 요소 트리에 대해 "전역"으로 간주될 수 있는 데이터를 공유하기 위해 Context API를 제공합니다.

```
setState(prevState => {  
  // Object.assign would also work  
  return {...prevState, ...updatedValues};  
});
```



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



참조 투명성 및 정적 유형 검사

JavaScript는 다중 패러다임, 객체 지향 프로그래밍 및 함수형 프로그래밍이며 React는 두 가지 모두의 강점을 물려받았습니다. 그러나 Hooks를 사용하면 함수형 프로그래밍에 더 편향된 것 같습니다.

OOP	FP
Inheritance <ul style="list-style-type: none">• Side effect• Statefull	Composition <ul style="list-style-type: none">• Pure• Stateless
Imperative (uses statements that change a program's state)	Declarative (expresses the logic of a computation without describing its control flow)



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



React의 핵심 기능은 구성 요소의 구성입니다. 서로 다른 사람들이 작성한 구성 요소는 함께 잘 작동해야 합니다. 코드베이스 전체에 파문을 일으키지 않고 구성 요소에 기능을 추가할 수 있다는 것이 중요합니다.

[React 구성 요소에서 부작용을 추출](#) 하면 더 예측 가능해집니다. 입력이 동일한 경우 구성 요소가 동일한 출력을 렌더링할 것으로 예상할 수 있습니다. 보다 구체적으로 말하면 참조 투명성을 얻거나 멍등성이 될 수 있습니다. 실제로 참조 투명성은 정적 유형 검사와 충분한 단위 테스트를 통해 보장되어야 합니다. 정적 유형 검사기 및 linter, 내가 선호하는 TypeScript 및 ESLint는 코드를 실행하기도 전에 특정 유형의 문제를 식별할 수 있으므로 개발 환경을 보다 확실하고 견고하게 만듭니다. 새 프로젝트를 시작할 때 구성이 다소 번거로울 수 있지만 귀하와 귀하의 팀이 훨씬 더 생산적인 데 도움이 됩니다. 2020년에 사용하지 않을 이유가 없습니다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



구성 요소 테스트

선언적 구성 요소는 구성 요소의 순수한 상호 작용 및 렌더링에 집중할 수 있으므로 테스트를 작성하기가 더 쉽습니다. Hooks를 사용하면 구성 요소에서 상태 저장 논리를 추출하여 독립적으로 테스트하고 재사용할 수 있습니다. 후크를 사용하면 구성 요소 계층 구조를 변경하지 않고도 상태 저장 논리를 재사용할 수 있습니다.

- [React Native Testing Library](#)가 이제 React Native를 위한 사실상의 테스트 라이브러리라고 입니다 . Jest와 밀접하게 통합되며 React Navigation 및 Redux와 같은 인기 있는 라이브러리와 함께 명확한 테스트 방법론을 제공합니다.
- [React Test Renderer](#) 는 React 코어와 함께 개발되었습니다. DOM이나 기본 모바일 환경에 의존하지 않고 React 구성 요소를 순수 JavaScript 객체로 렌더링합니다.
- [React Native Testing Library](#) (또는 RNTL)는 React Test Renderer 위에 구축됩니다. 렌더링(to getByText, queryAllByA11yRole, ...), fireEvent, waitFor 및 act와 같은 유용한 API를 추가합니다. 사용자 경험과 접근성에 초점을 맞추는 데 의견이 분분합니다.
- [React Hooks Testing Library](#)는 구성 요소 상호 작용을 통해 테스트하기 어려운 구성 요소 또는 컴플렉스에 직접 연결되지 않은 테스트 사용자 지정 후크를 작성합니다.



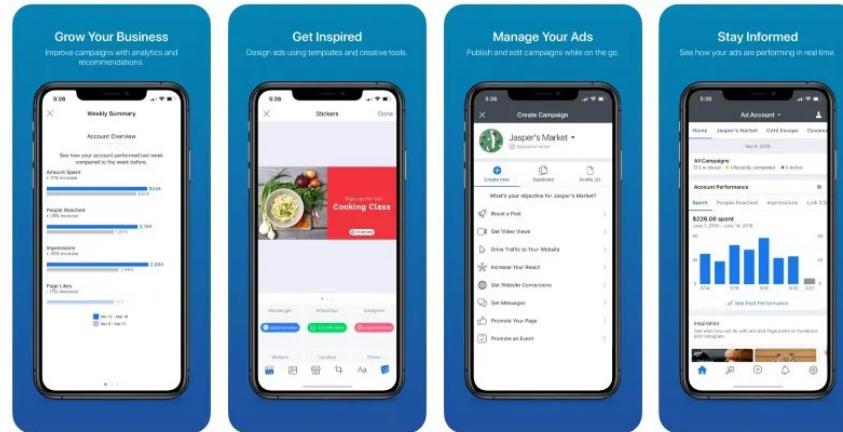
Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



Facebook 광고 관리자 및 React Native

Facebook은 기본 애플리케이션에서 React Native를 사용한 후에도 멈추지 않았습니 다. 가장 좋은 예의 또 다른 하나는 실제로 React Native만을 기반으로 하는 최초의 크로스 플랫폼 앱인 [Facebook Ads Manager](#) 입니다 .
광고 형식, 시간대, 날짜 형식, 통화 및 사용자마다 다른 많은 것들을 처리하는 데 필요 한 비즈니스 로직 때문에 복잡한 작업이었습니다. 그러나 대부분의 코드가 JavaScript로 작성되었기 때문에 React Native가 이상적이었습니다.
Facebook 개발 팀이 Facebook 광고 관리자를 작업하는 동안 다른 개발자에게도 유 용한 많은 구성 요소를 만들었습니다.



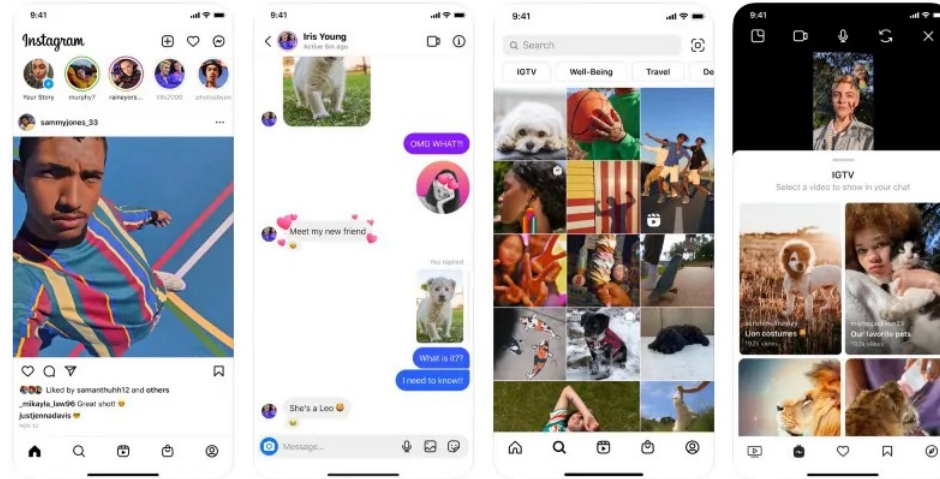
Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



인스타그램과 리액트 네이티브

React Native Apps의 또 다른 인기 있는 예는 Instagram입니다.

처음부터 전체 앱을 구축하는 것보다 더 어려운 것은 새로운 기술을 기존 앱에 통합하는 것인데 Instagram이 시도한 것은 바로 그것이었습니다. 그들은 상상할 수 있는 가장 단순한 보기인 푸시 알림 보기에서 시작했습니다. 처음에는 내비게이션 인프라를 구축할 필요가 없었기 때문에 WebView로 구현되었습니다.

React Native는 Instagram 개발자가 **Android** 및 **iOS** 앱 모두에 훨씬 빠르게 기능을 제공할 수 있도록 했습니다.



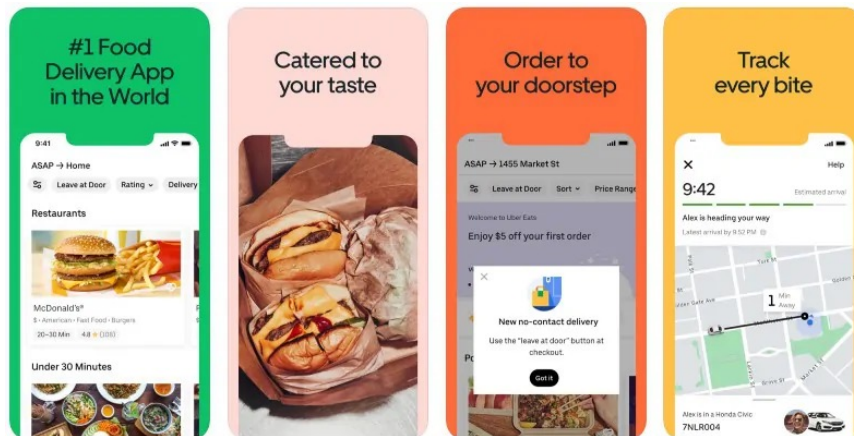
Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



UberEats와 리액트 네이티브

표준 Uber 애플리케이션에는 운전자와 승객이라는 두 당사자가 포함되지만 UberEats에는 식당, 배달 기사 및 고객의 세 당사자가 포함됩니다. 이러한 모델은 레스토랑 전용 대시보드가 추가로 필요했습니다.

Uber는 웹을 염두에 두고 원본 대시보드를 만들었고 소리 알림과 같은 기본 기능에 대한 액세스가 제한되었습니다. 결과적으로 사용자 경험이 약해졌습니다.

Uber 개발팀은 React Native를 사용하여 전체 대시보드를 재구축하기로 결정했습니다. UberEats 앱의 전체 코드 중 작은 부분에 불과하지만 개발자는 React Native가 향후 시장이 성장함에 따라 새로운 기능을 구현하는 데 도움이 될 것이라고 낙관하고 있습니다.



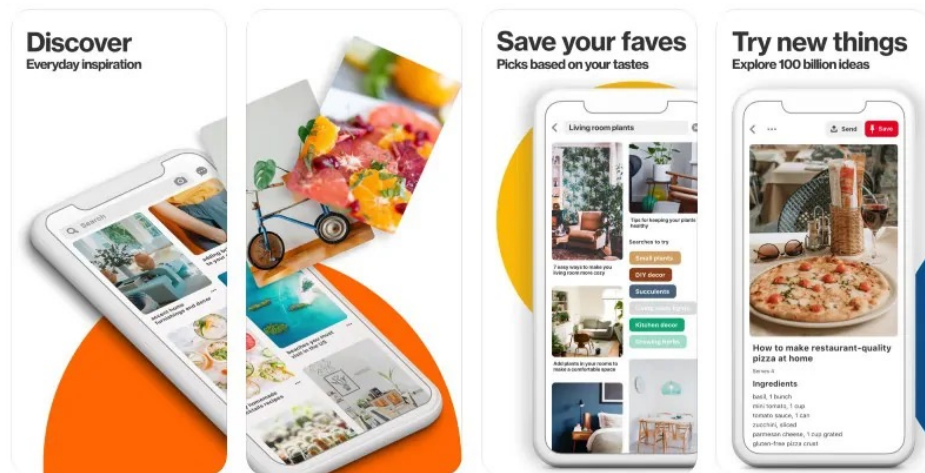
Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



핀터레스트와 리액트 네이티브

Pinterest는 이를 기술 스택에 구현하기 위해 React Native를 고려했고, 그들이 찾은 이점에 따라 그렇게하기로 결정했습니다. 가장 큰 것은 개발자 속도의 증가였습니다. React Native를 사용하면 플랫폼 간에 코드를 공유할 수 있으므로 스탠드업 및 회의에 소요되는 시간이 줄어들고 모바일 플랫폼 간의 불일치가 줄어듭니다.

그러나 Pinterest는 Android 및 iOS의 기본 코드 기능 옆에 있는 엔지니어링 도구 상자의 추가 도구로 React Native를 사용합니다. 그들은 전체 코드를 React Native로 다시 작성하고 싶지 않았습니다.

Pinterest는 React Native를 사용하여 주제 선택기 보기 및 비즈니스 가입 흐름을 구축했습니다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



수업진행방식

1. 내용소개 및 배경설명 그리고 학습목표 제시
2. 코드 구성 및 설명
3. 예제 그대로 따라하고 이해
4. 실습 과제 제시
5. 실습
6. 실습 과제 제출



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

1st Lecture

Mobile Computing



Q&A