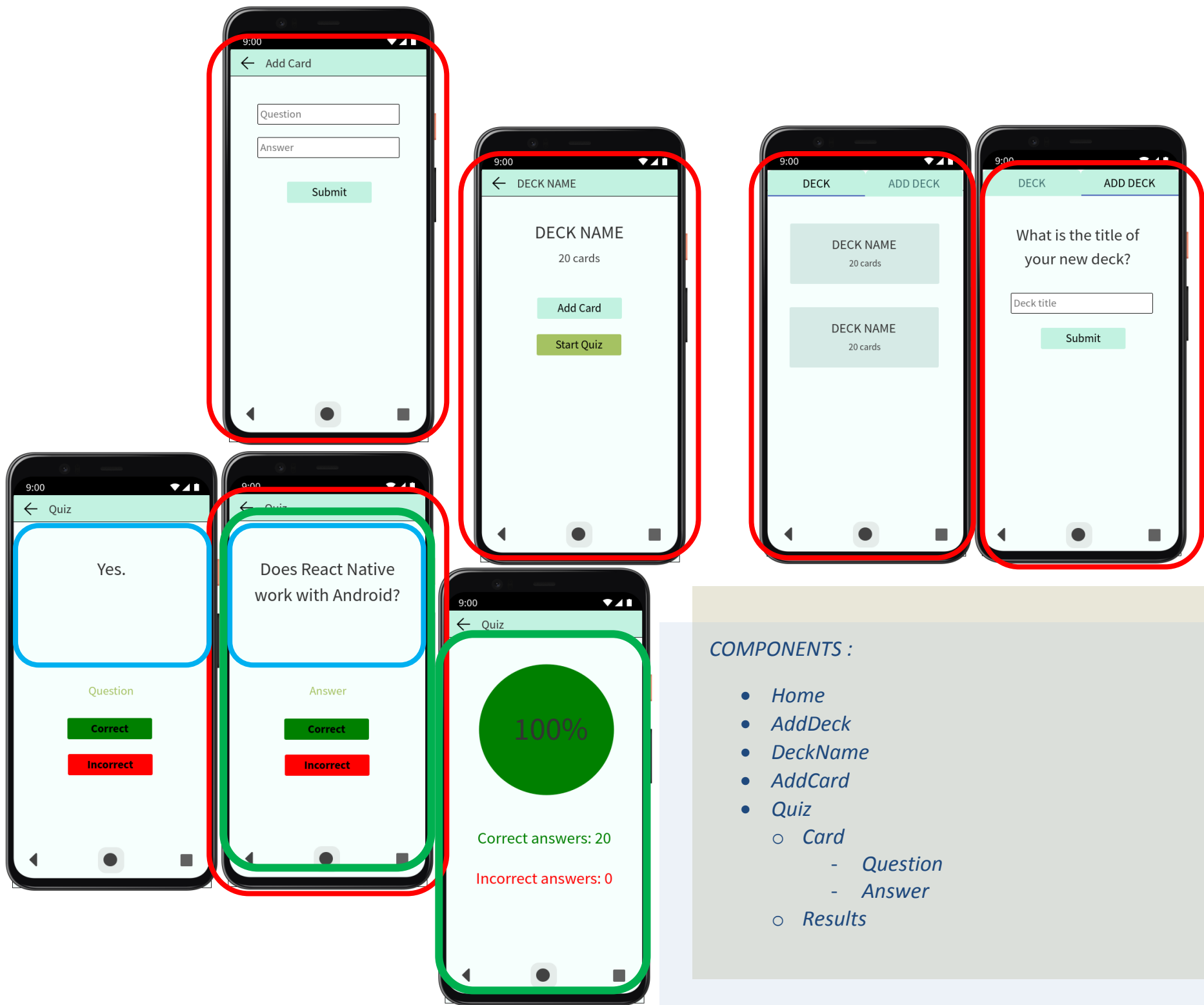


# Flashcards App

Created on May 24 2020 by Jaka Presecnik

I'll try to take a different approach on this project. What I'll try to do, is create as rich process for creating the app in Word as possible. If I'll be motivated during the coding stage I'll comment what was different than planned.



### COMPONENTS :

- Home
- AddDeck
- DeckName
- AddCard
- Quiz
  - Card
    - Question
    - Answer
  - Results

## 1.0. Events in the app:

### 1.1. *Get data and update UI:*

Use `componentDidUpdate` to get the data from the store

### 1.2. *Submit a new deck and update state*

Inside AddDeck component create a `state` to store the name from input for new deck's name, and create `onSubmit` function that sends the data to the store.

### 1.3. *Allow user to interact with individual deck*

Create `removeDeck` function that allows user to remove the deck from collection and update the store onSubmit using `onSwipeRight`.

Commit before continuing!

Use:

```
npm i -S react-native-swipe-gestures
```

And follow instructions on <https://www.npmjs.com/package/react-native-swipe-gestures>, if not satisfied and if this was easy to do, reset commit and follow <https://software-mansion.github.io/react-native-gesture-handler/docs/component-swipeable.html>, which adds a different functionality on swiping.

Decide on what you'll use: the `onPress` or `onSwipe` methods to open an individual deck screen.

### 1.4. *Open a particular deck from the state*

1.4.1. *Add a new card and update the state*

1.4.2. *Show the question*

1.4.3. *Show the answer*

1.4.4. *Submit the answer to the state and to the store*

1.4.5. *Show the results for the session*

## 2.0. Data in the store:

Data for individual deck:

```
{
  key: generateId(),
  name: 'deckName',
  collection: [{
    key: generateId(),
    question: 'question',
    answer: 'answer',
    answeredCorrectNum = 0,
    answeredIncorrectNum = 0,
  }]}
}
```

## 3.0. Data in Components:

### 3.1. *AddDeck:*

- `deckName` from input

### 3.2. *AddCard:*

- `question` from input
- `answer` from input

### 3.3. *Quiz:*

- `cardNum = 0`
- `answerSide = false`
- `correctNum = 0`
- `incorrectNum = 0`