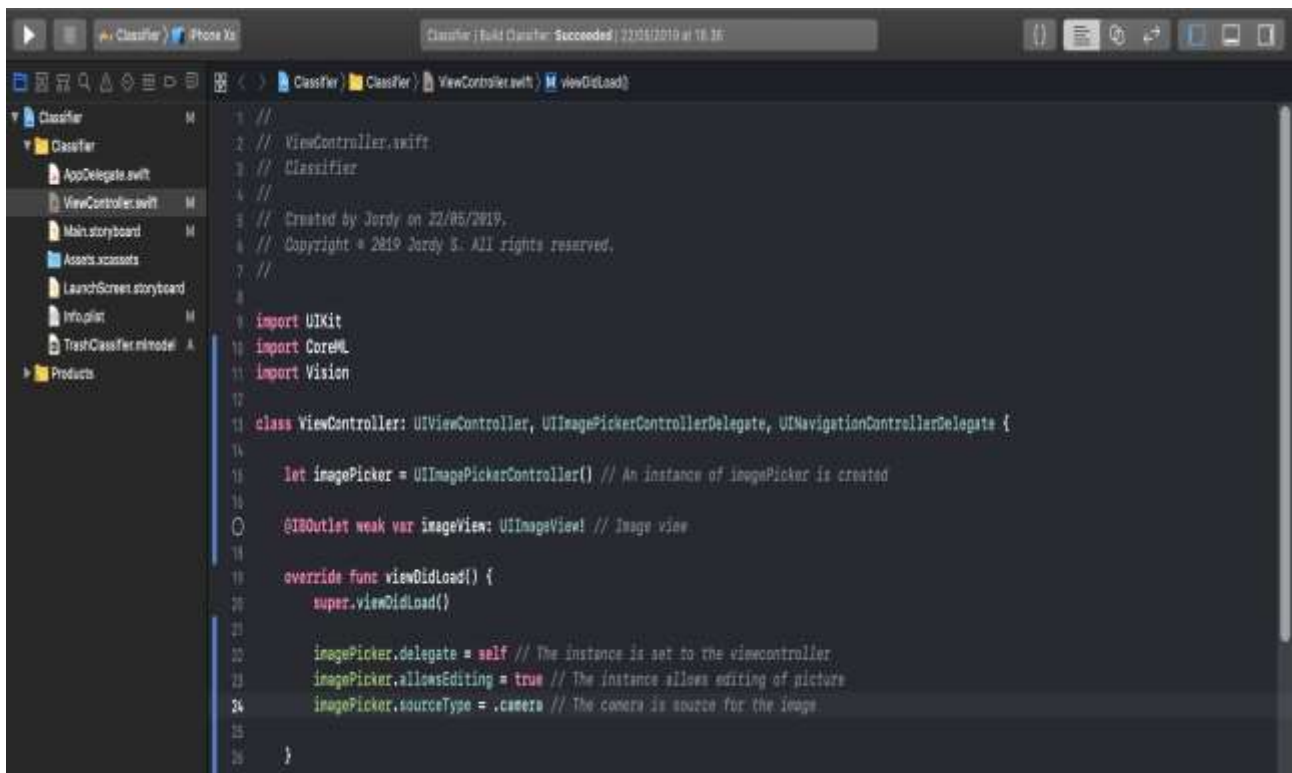
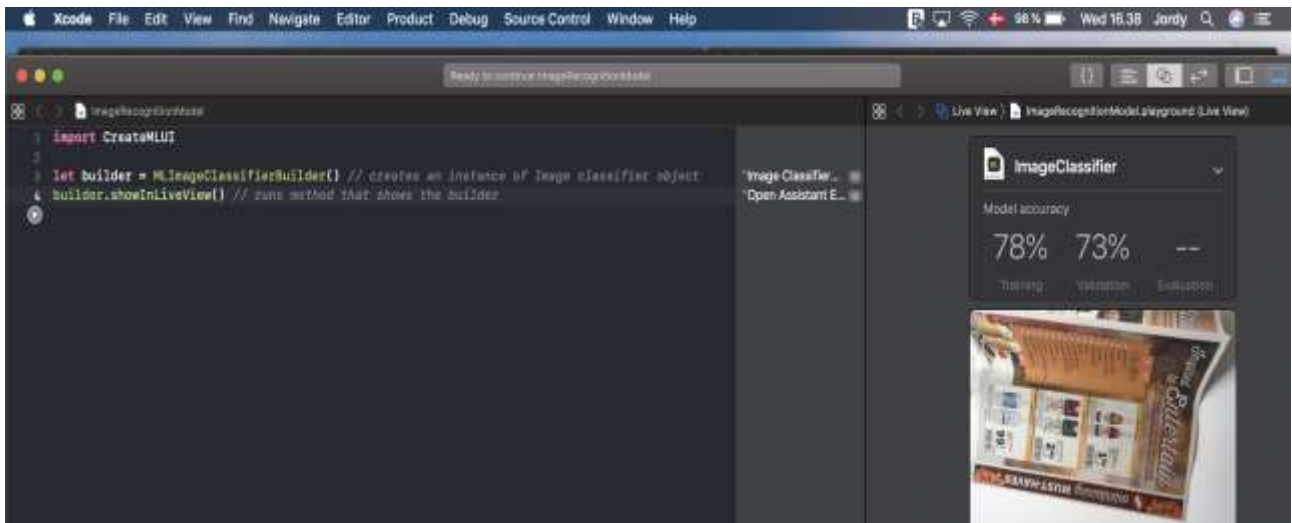


Produkt: koder



```
Classifier | Built Classifier: Succeeded: 22/08/2019 at 16:36
Classifier | Classifier | ViewController.swift | viewDidLoad

28
29
30 func imagePickerController_ picker: UIImagePickerController, didFinishPickingMediaWithInfo info: [UIImagePickerController.InfoKey : Any]){
31
32     if let userPickedImage = info[UIImagePickerController.InfoKey.editedImage] as? UIImage { // If user picks image and it can be converted UIImage
33
34         guard let convertedUIImage = UIImage(image: userPickedImage) else { // Then call UIImage and convert UIImage userPickedImage to UIImage
35             fatalError("Cannot convert to UIImage.")
36         }
37
38         detect(image: convertedUIImage) // convertedUIImage is passed in the method
39
40         imageView.image = userPickedImage
41
42     }
43
44     imagePicker.dismiss(animated: true, completion: nil)
45 }
46
47
48 func detect (image: UIImage) { // Function detects takes parameter image in the format of UIImage: containsUIImage
49     // Vision container for the model
50     guard let model = try? VNCoreMLModel(for: TrashClassifier().model) else {
51         fatalError("Cannot import model")
52     }
53
54     // Request
55     // when the request has been completed
56     // The completion handler will activate and we'll get request or error
57     let request = VNCoreMLRequest(model: model) { (request, error) in
58         let classification = request.results?.first as? VNClassificationObservation // Process the request
59
60         self.navigationItem.title = classification?.identifier.capitalized // Pass on the navigation bar to show classification
61     }
62
63     // handler
64
65     let handler = VNImageRequestHandler(cUIImage: image)
66     do {
67         try handler.perform([request])
68     }
69     catch {
70         print(error)
71     }
72 }
73
74 @IBAction func cameraTapped_ sender: Any {
75     presentImagePicker, animated: true, completion: nil)
76 }
77
78 }
```

