Michal Bochenek  
EC1401916

Multimedia:   
**Developing Multimedia Applications DH2R34/004**  
  
Outcome 4:   
**Test and evaluation report.**

Table of Contents

[1. Introduction and evaluation of the final design. 3](#_Toc452474266)

[2. Structure chart evaluation – navigation. 4](#_Toc452474267)

[3. Storyboards (using Google Chrome mobile view). 5](#_Toc452474268)

[4. Application - prototype code example (Notepad++). 7](#_Toc452474269)

[5. Test plan and test runs. 8](#_Toc452474270)

# Introduction and evaluation of the final design.

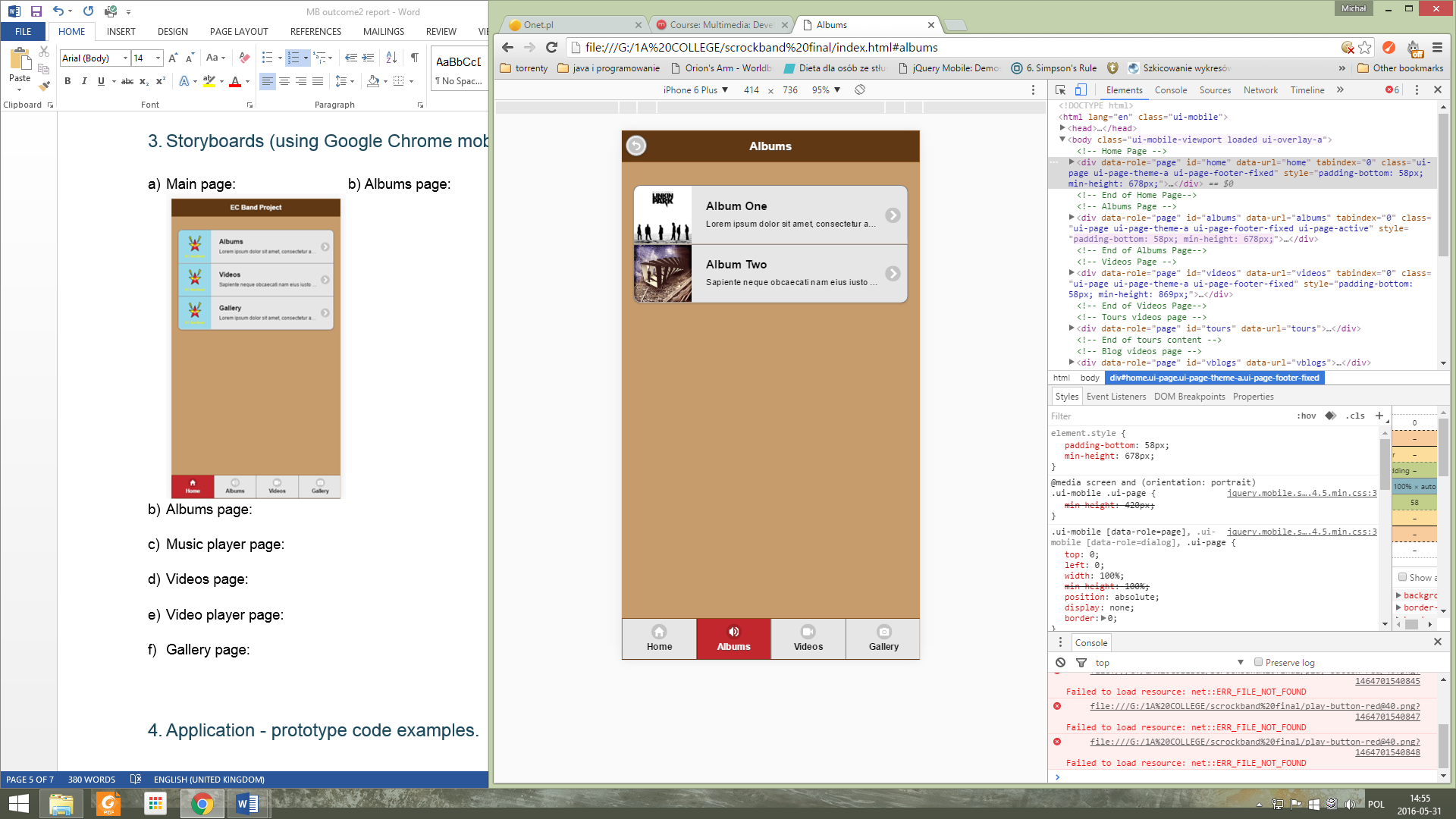
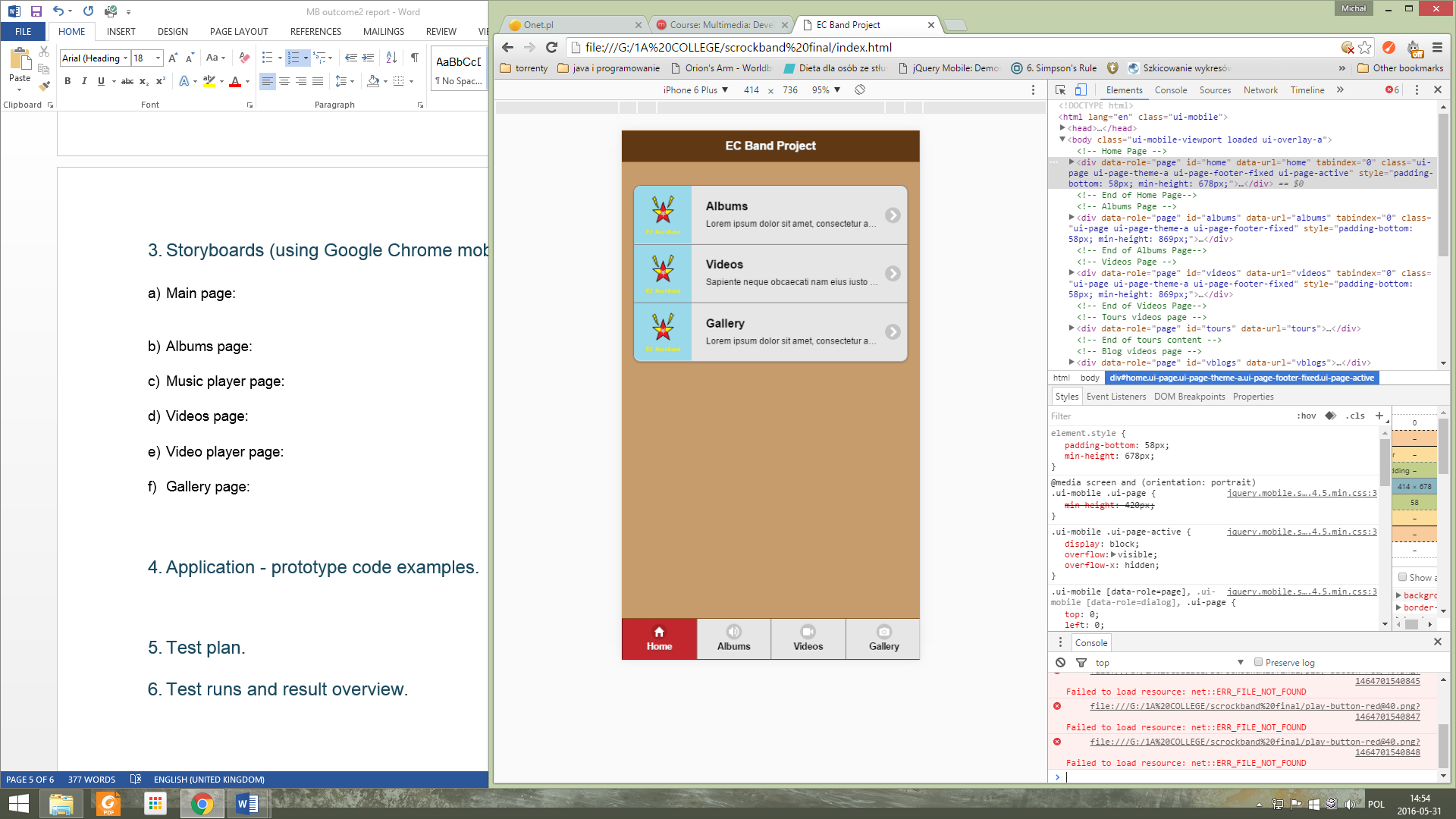
This document is a summary of the development and implementation stage of the multimedia project for the EC RockBand mobile web application.  
It contains report on design and navigation scheme, evaluation of it and test plan with test runs and shown results.

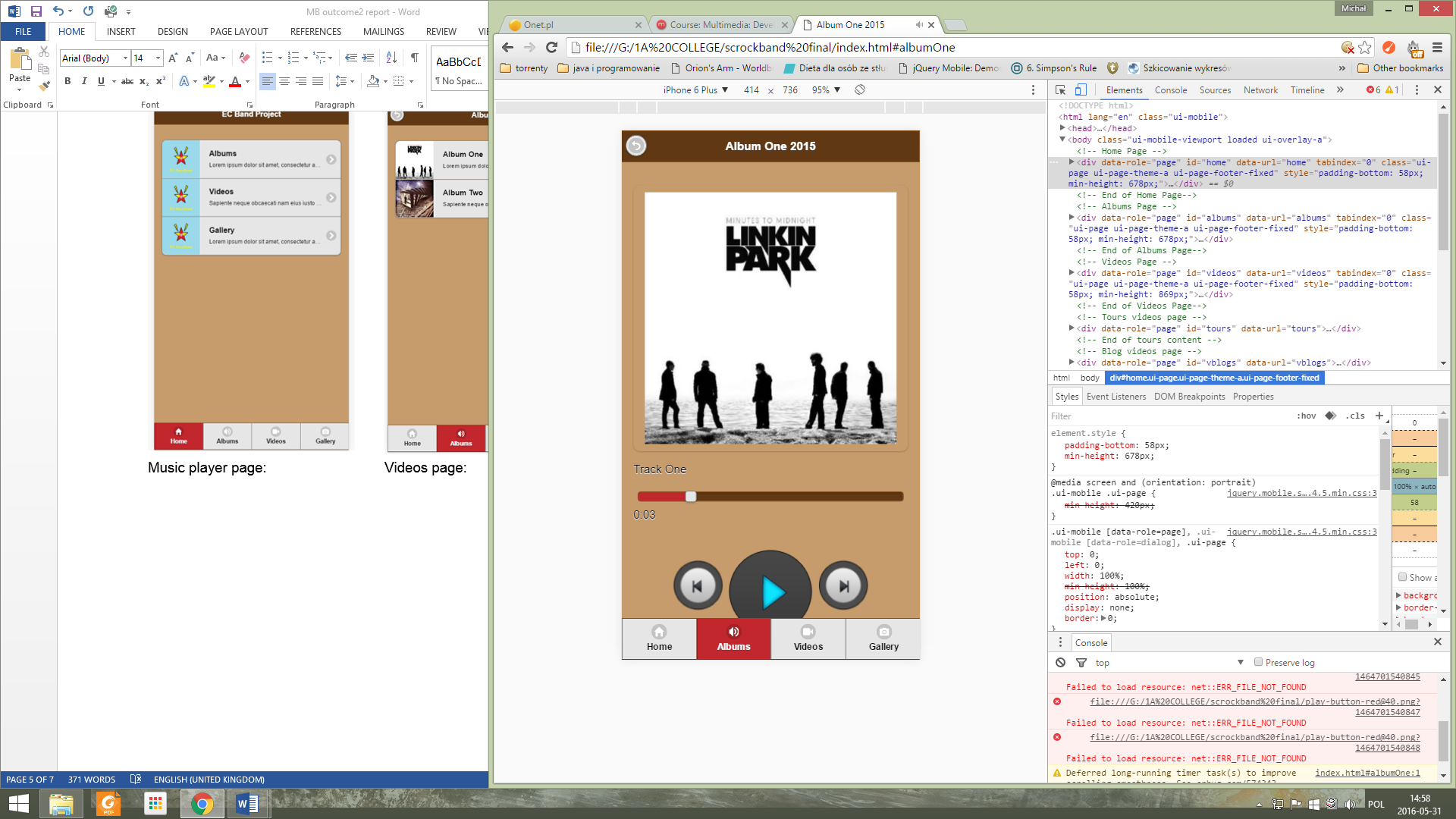
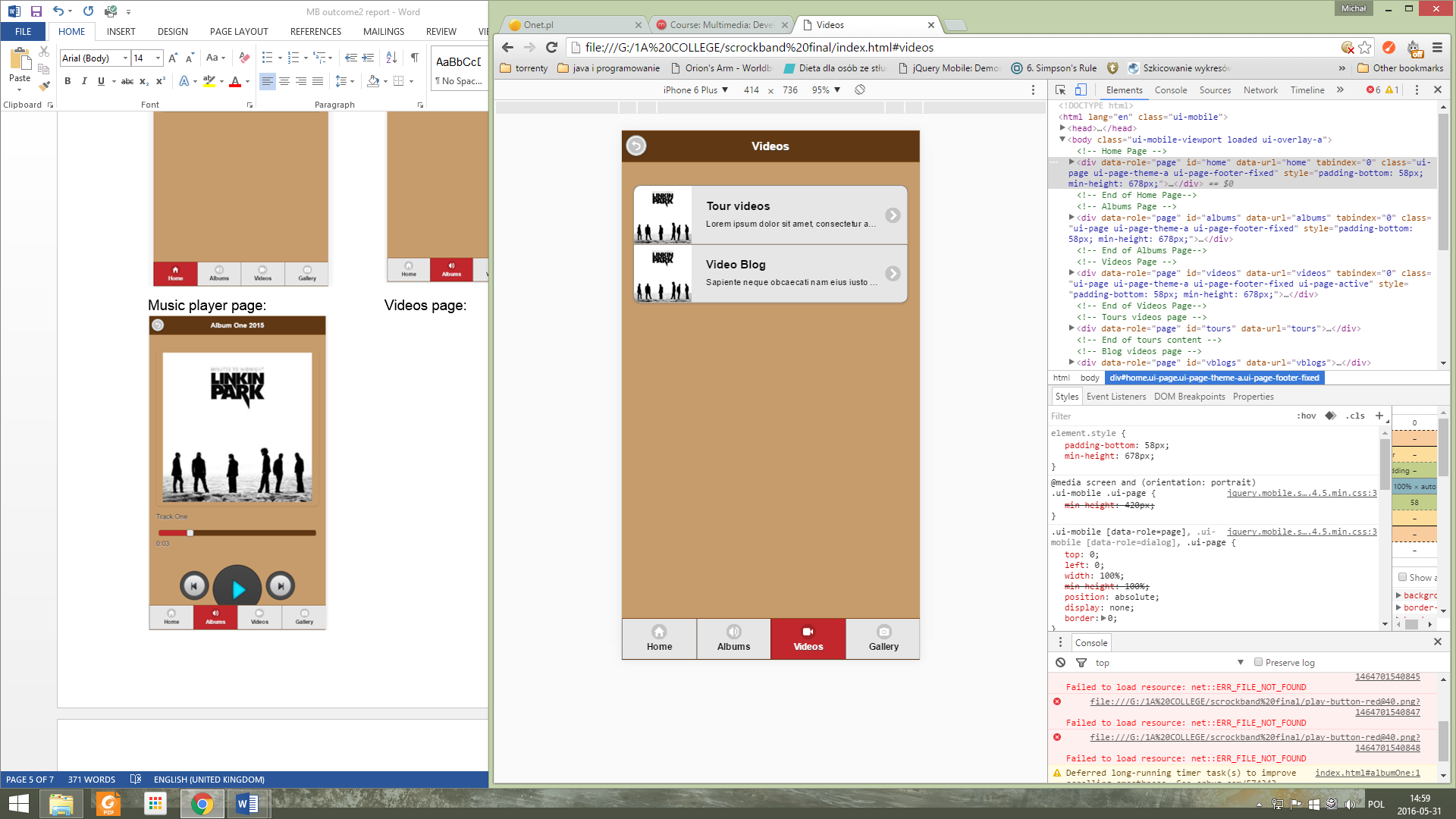
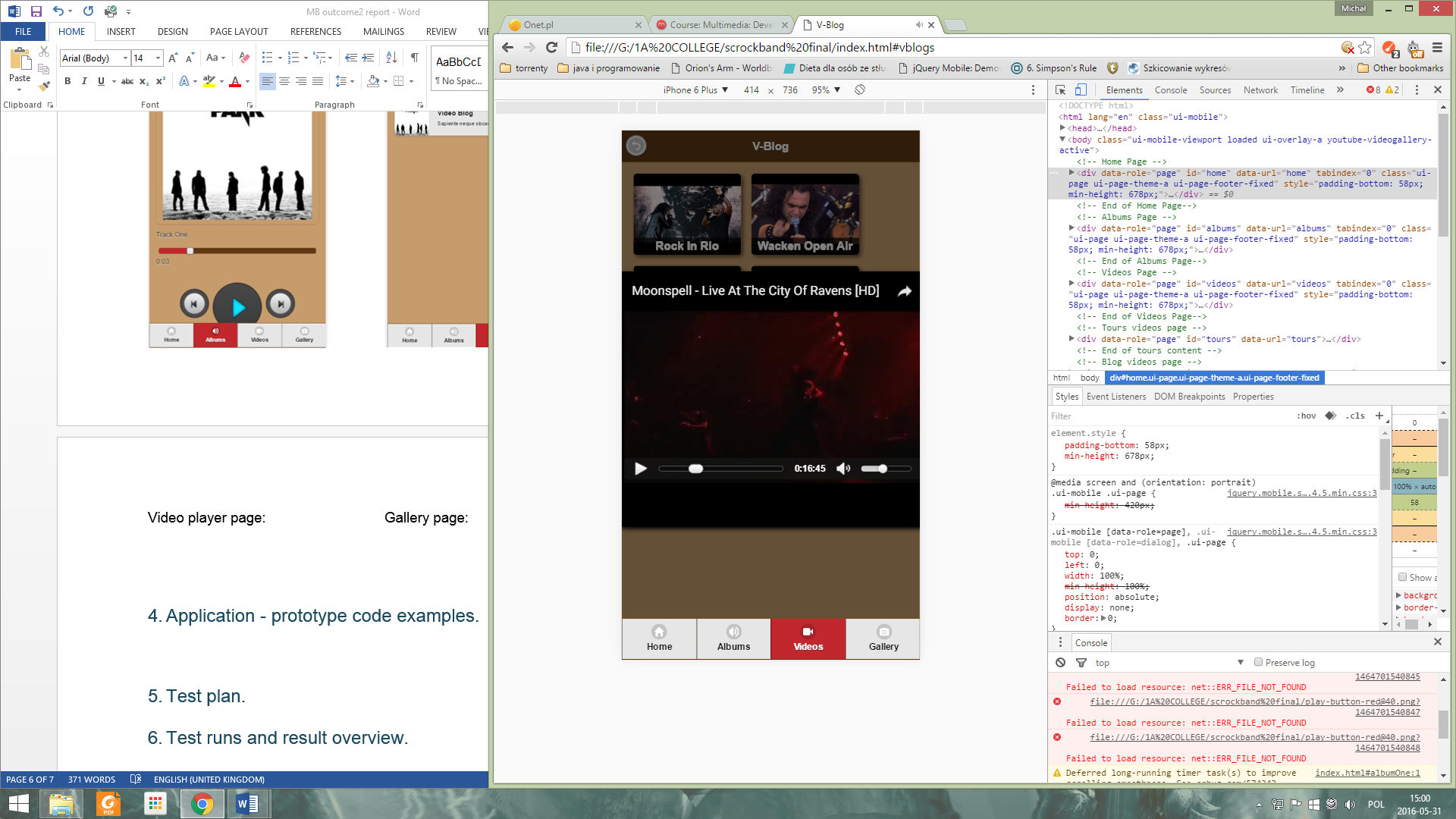
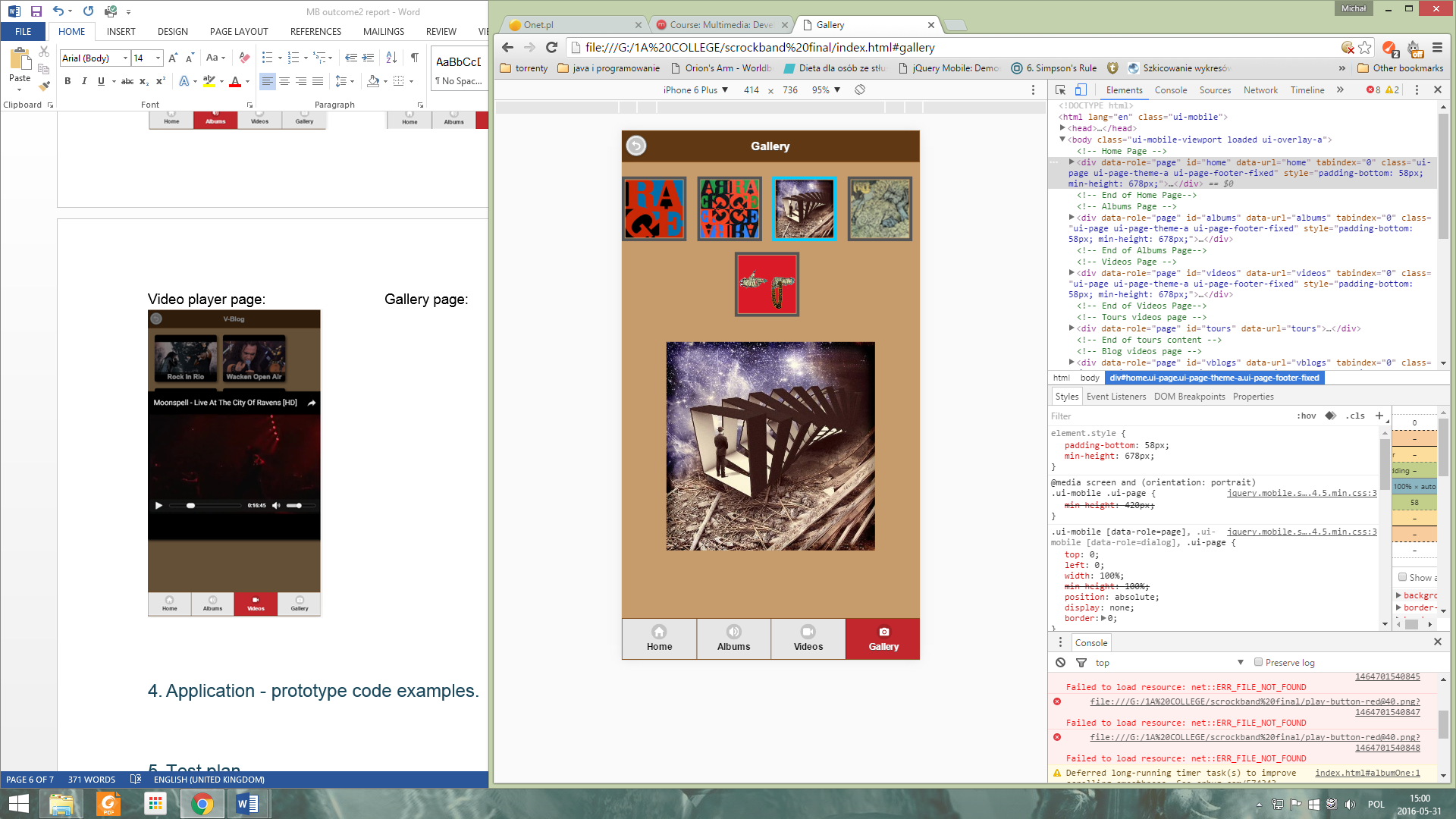
The final application consists of the components such as:  
  
a) **Home page area** – this is the main page for the application to which user is being redirected on start and from which navigation to the content begins.  
  
b) **Albums page area** – in this content area a user is given a choice of EC RockBand recorded music albums containing sample audio files.  
  
c) **Album player area** – this page is displayed after user selects album folder and from there on he/she is being redirected here where music files grouped into a list can be played using in-built player.   
  
d) **Videos page area** – in this content area a user is given a choice of EC RockBand YouTube video records grouped into thematically selected albums.  
  
e) **Video player area** – this page is shown once a user selects the video folder and it uses a specially designed plugin for displaying YouTube videos on small devices without the need of using a browser or external application.  
  
f) **Picture gallery area** – this part of the page loads selected pictures thumbnails and displays them enlarged once user clicks/taps on the link.  
  
  
  
  
  
  
Overall user requirements specification was met and the application does what it was intended in the initial client brief, a proof of which is shown in the following points.  
   
Due to the time constraints a number of changes into initially included in the project plan features, was necessary to be done, such as:   
removal of a user registration, logging and account management and addition in place of a tour calendar and mini blog - a picture gallery.  
In future releases these contents can be reintroduced and existing contents can be upgraded and improved based on user feedback.  
The client (the lecturer) reviewed and agreed upon these changes.

# Structure chart evaluation – navigation.

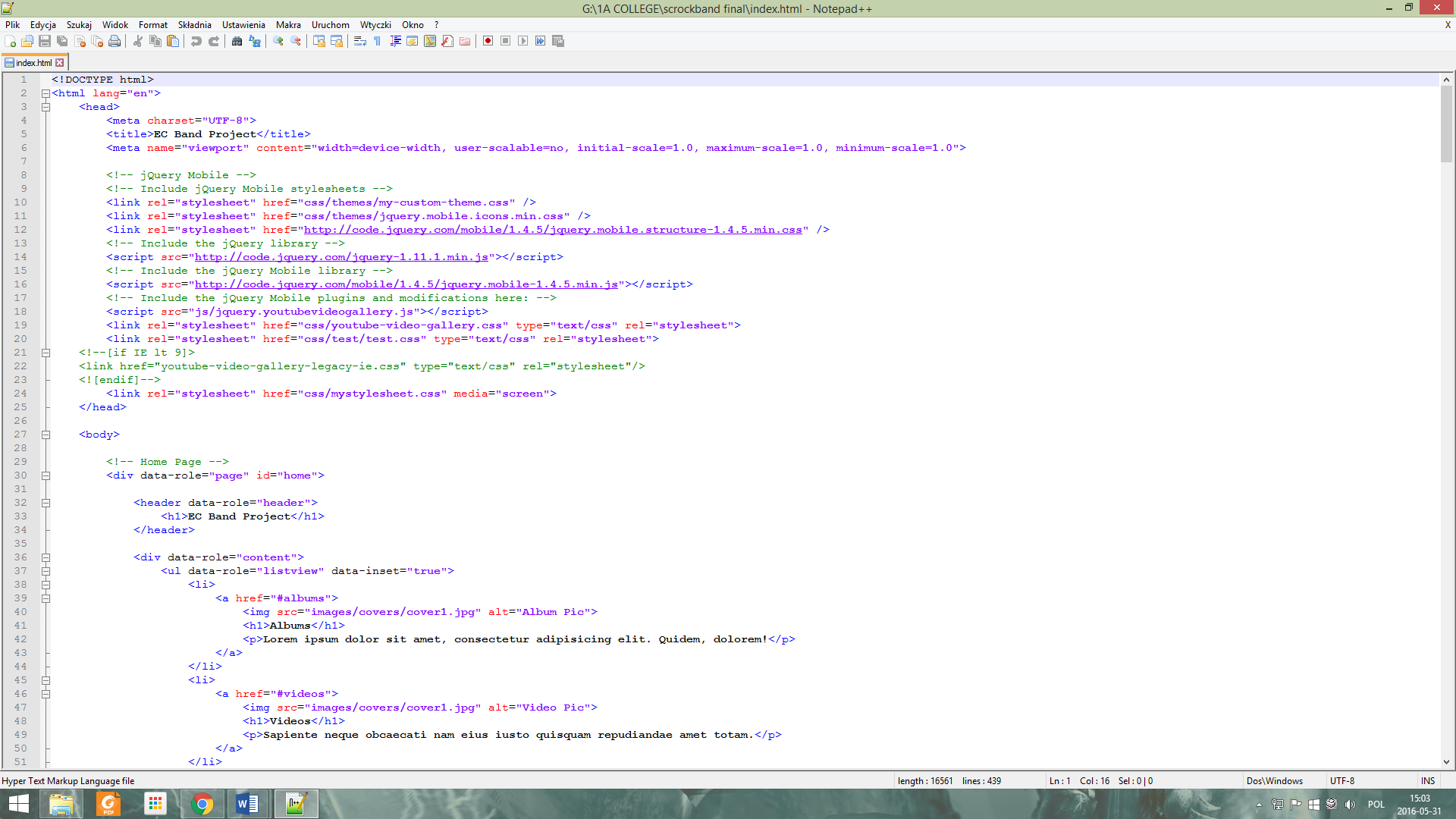
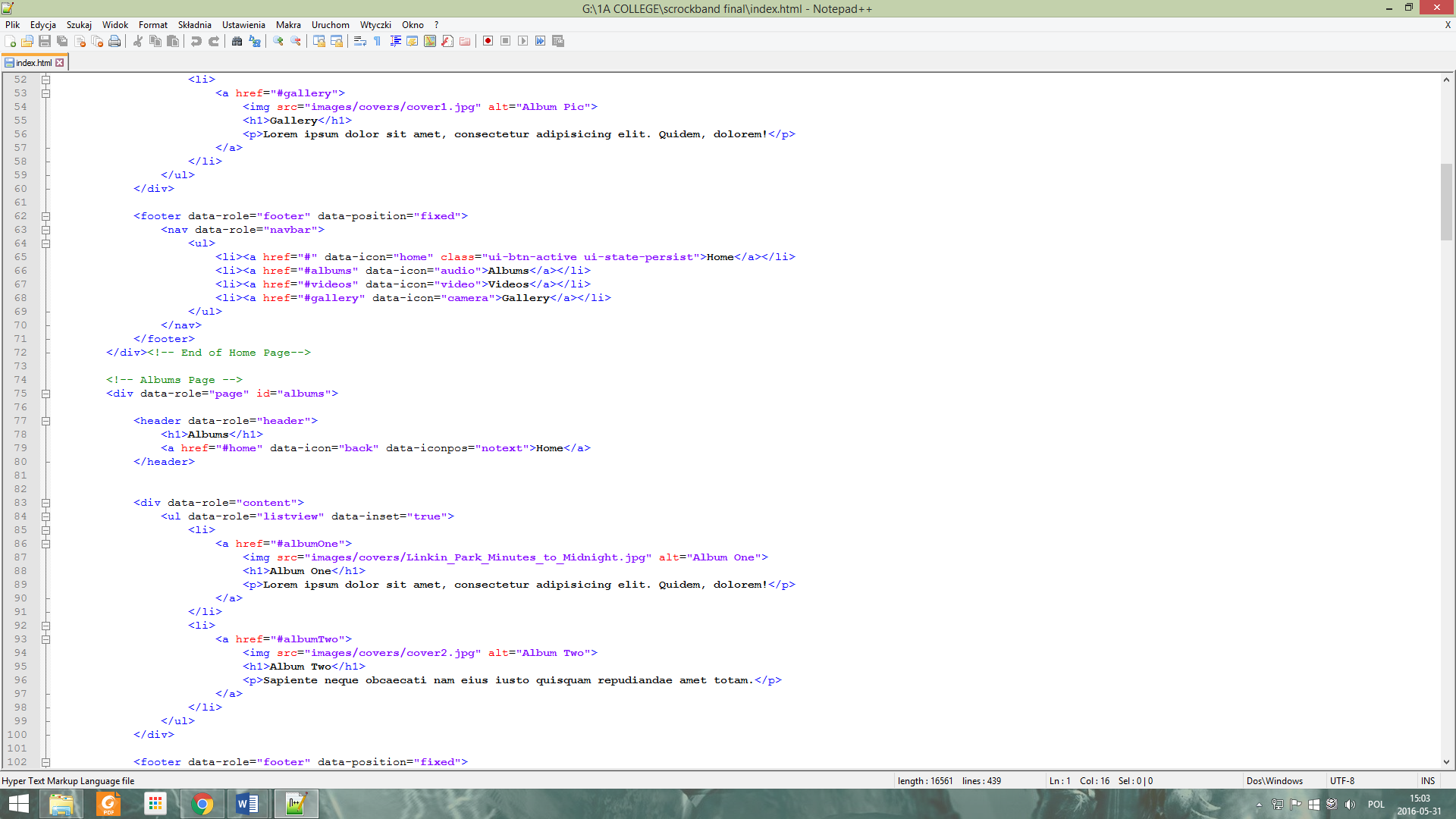
The application navigation scheme was simplified to reduce the learning curve to minimum and loading time spent on getting to the content of interest by the user.   
  
User can go back in the hierarchy by pressing return button which was placed on the top-left side of the header for each page.

# Storyboards (using Google Chrome mobile view).

Main page: Albums page:  


Music player page: Videos page:  
 Video player page: Gallery page:  
 

# Application - prototype code example (Notepad++).

# Test plan and test runs.

USER ACCEPTANCE TEST PLAN:

FROM THE DEFINTION:   
“A testing technique performed to determine whether or not the software system has met the requirement specifications.” [Tutorialspoint, *n.d*]

INSTRUCTIONS TO THE TESTER:   
- open an up to date Google Chrome web browser  
- navigate to the web-app folder and start index.html  
- in the website window right-click on it and select “inspect”   
- change view to mobile device (iPhone6) and run the tests

POSSIBLE ISSUES PRIOR TO TEST:  
  
This test has one main issue – it can’t be used for a proper in-code debugging as it is a black box test type, an external tester doesn’t know the code and might need to refer to the code listing in order to check for the expected result or assume requirement specification as a pass/fail criteria.

TESTED ITEMS AND TEST RUNS LOG:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Date / Tester | Item to Test | Test Purpose | Expected Results | Actual Results | Comments |
| 05/05/16  Michal Bochenek | Main page | Meet the user requirements specification | Application starts and displays menu waiting for user input | Application started and displayed menu.  PASSED | Look for picture 1 below the table for output |
| 05/05/16  Michal Bochenek | Albums page | Meet the user requirements specification | Albums page displayed and loaded existing folders (2) | Albums page displayed and loaded both existing folders PASSED | Look for picture 2 below the table for output. |
| 05/05/16 Michal Bochenek | Music player | Meet the user requirements specification | Music player displayed and works correctly by responding on given commands (stop, start, next previous, select track from the list) | Player responded on all commands successfully   PASSED | Look for picture 3 below the table for output. |
| 05/05/16 Michal Bochenek | Videos page | Meet the user requirements specification | Videos page displayed and loaded existing video folders (2) | Video page displayed and loaded both existing folders  PASSED | Look for picture 4 below the table for output. |
| 05/05/16 Michal Bochenek | Video player | Meet the user requirements specification | Video player displayed and works correctly by responding on given commands using YouTube UI buttons. | Video player responded on all commands successfully   PASSED | Look for picture 5 below the table for output. |
| 05/05/16 Michal Bochenek | Gallery page | Meet the user requirements specification | Gallery page displayed with default picture selected (picture 1) and loaded existing sample pictures (5) | Page displayed and loaded all 5 existing sample pictures  PASSED | Look for picture 6 below the table for output. |

TEST APPROACH:  
  
To perform a test a tester has to follow the program menu instructions and provide the correct (expected) input.  
  
TEST PASS/FAIL CRITERIA:

Test passes if the tested item expected result is met with the actual output.  
Failure is when there is any other or no output displayed.

TEST ENTRY/EXIT CRITERIA:   
  
Entry criteria:

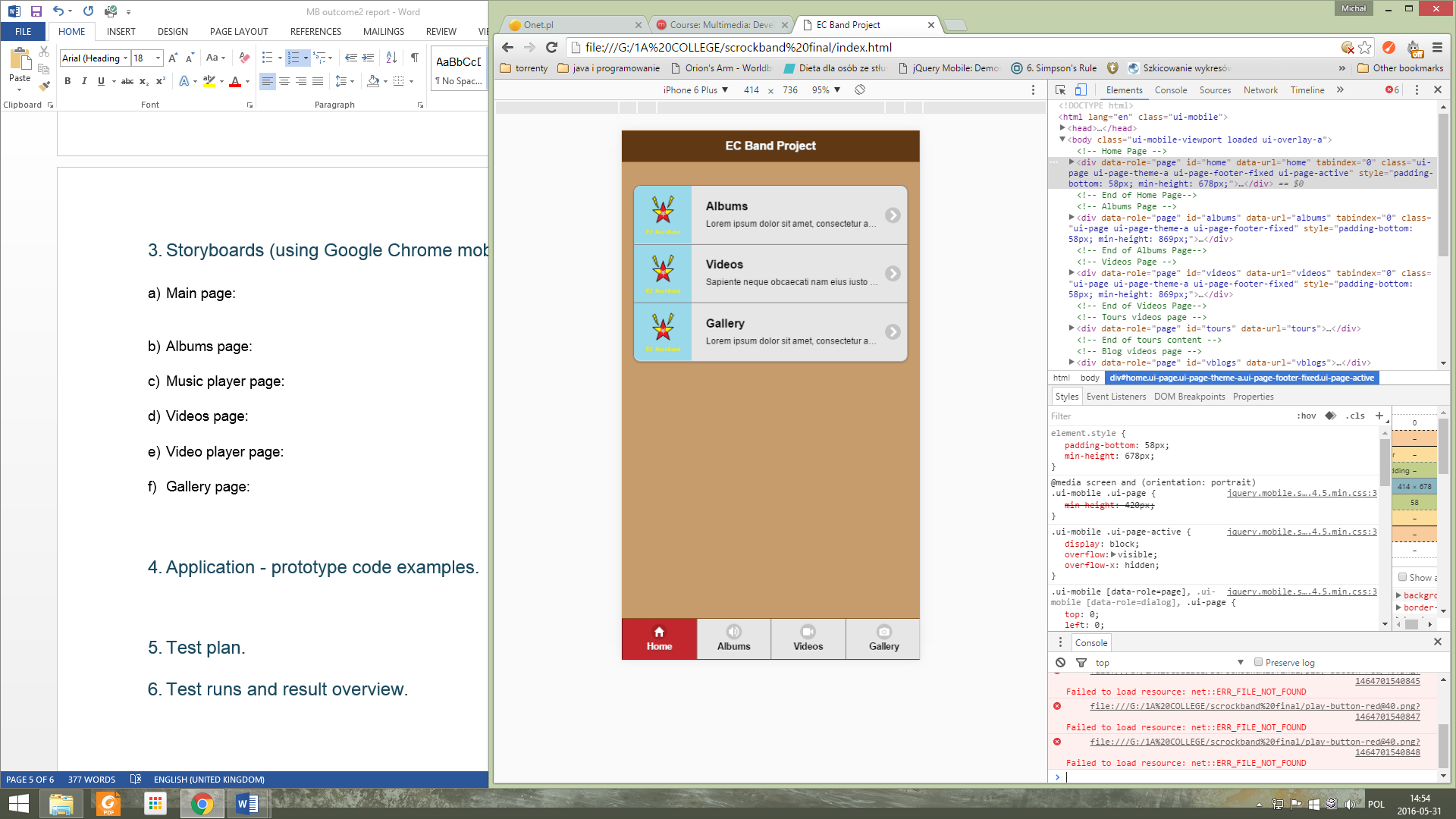
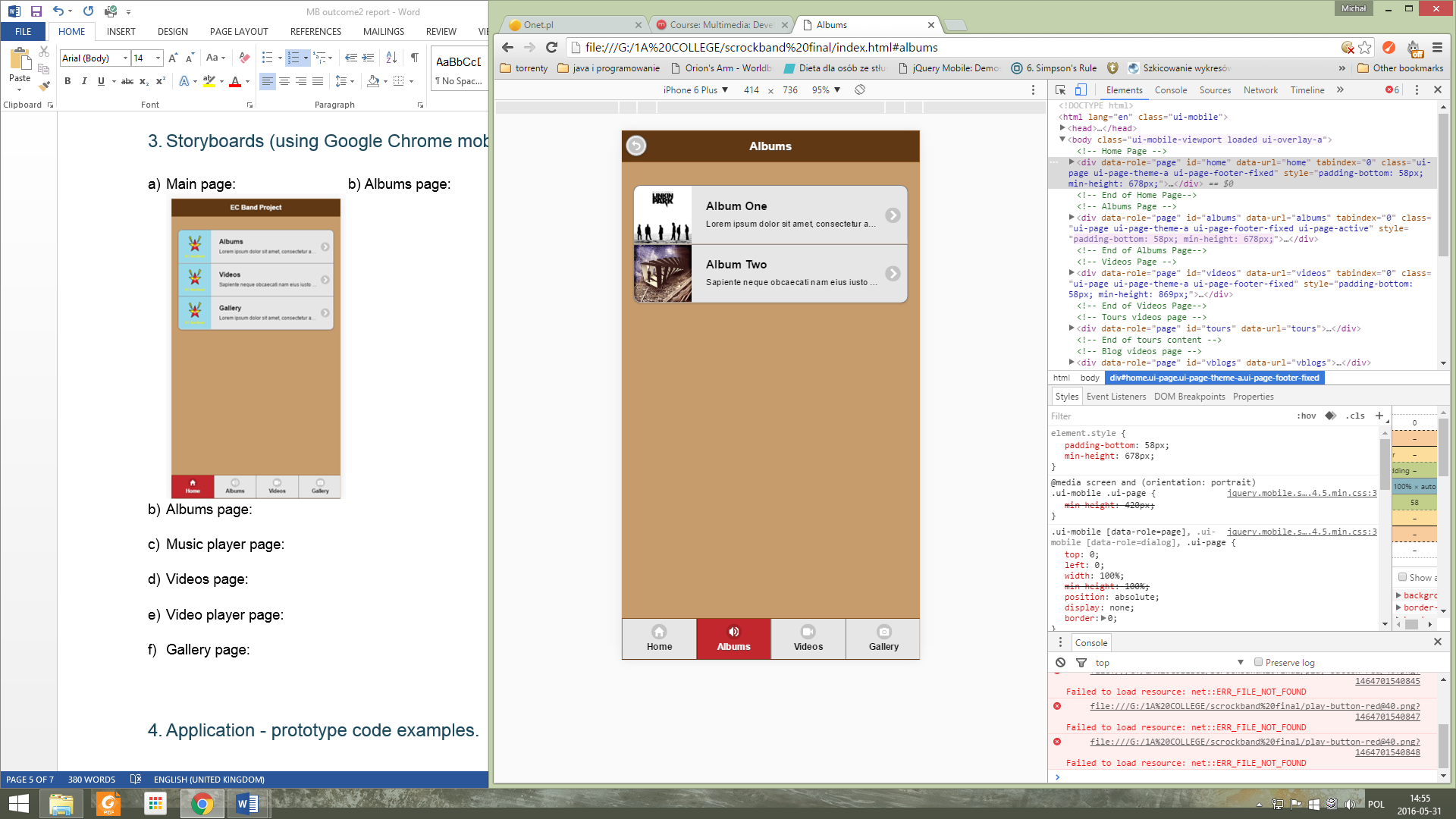
* Up to date Google Chrome browser
* Testable code is available
* Test data is checked and exists inside the root of a code folder
* Test data is correct

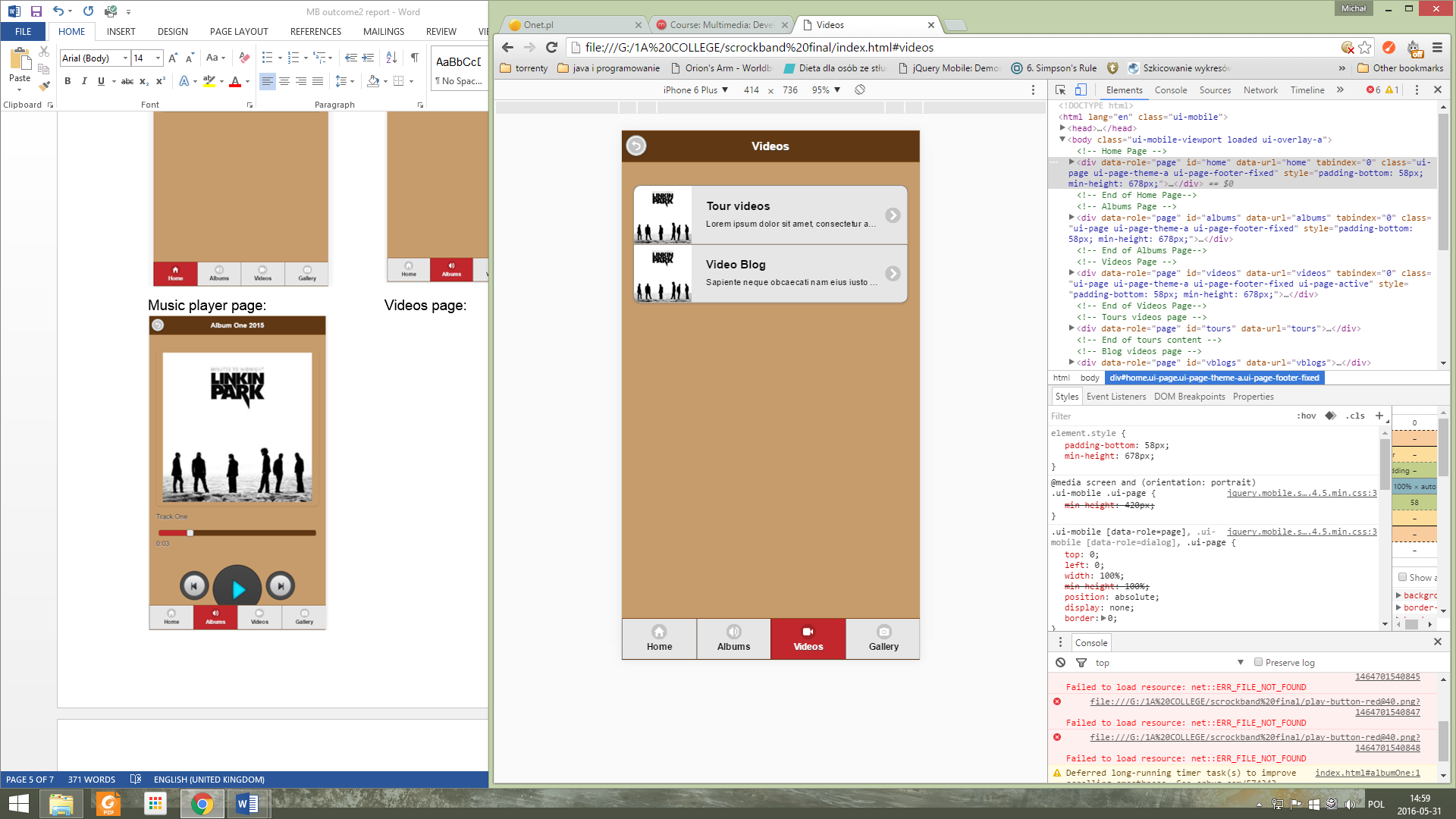
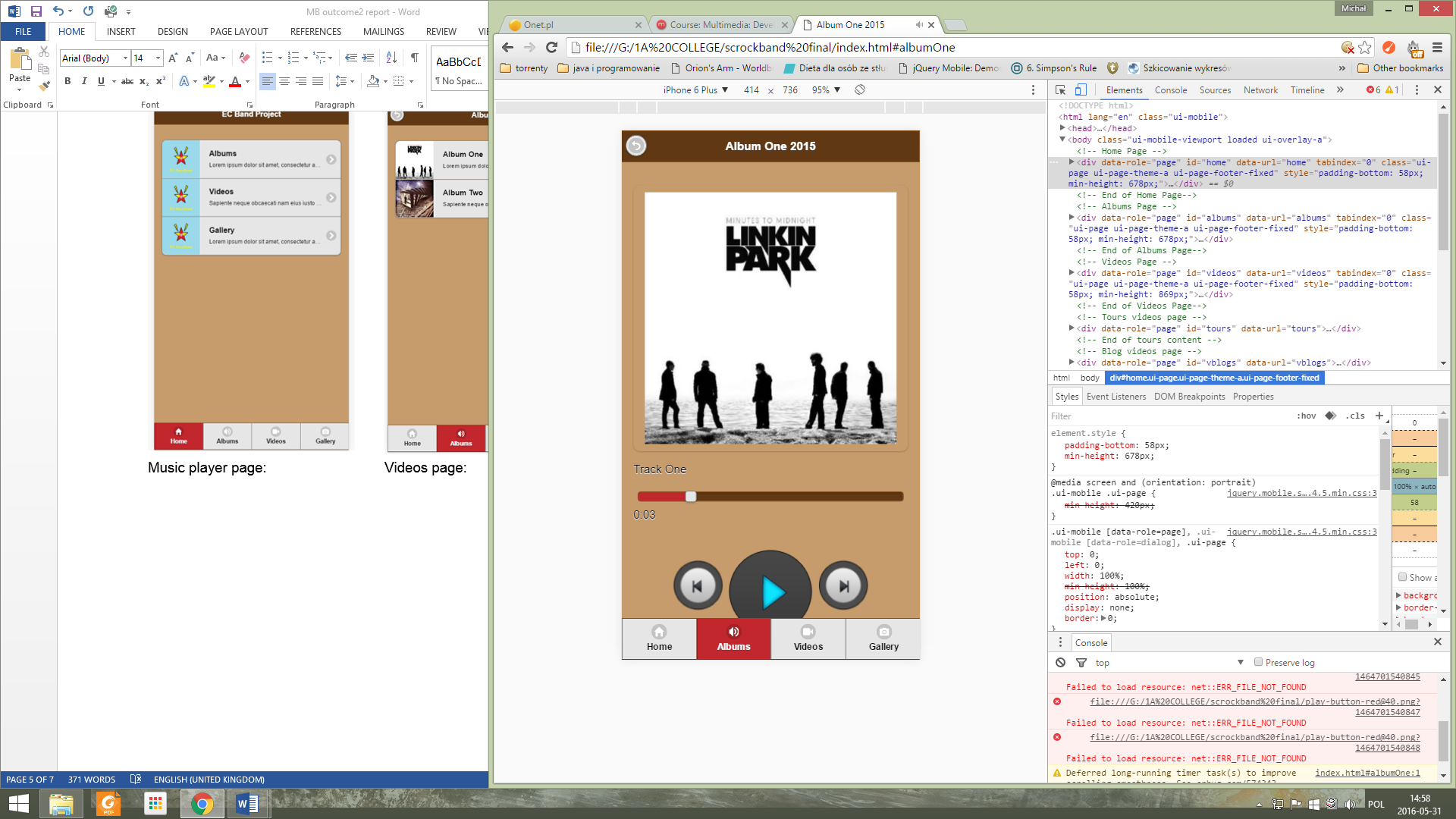
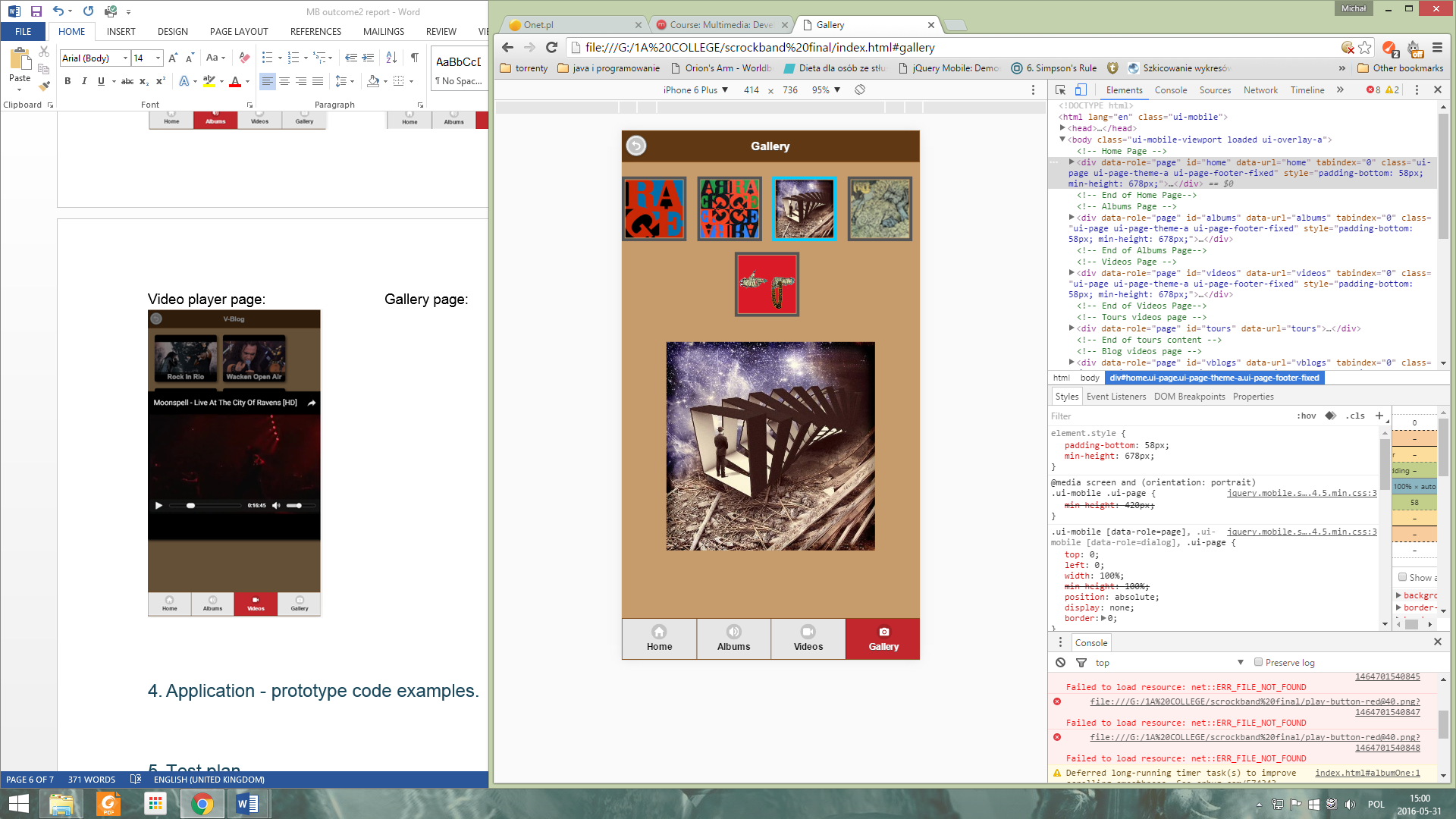
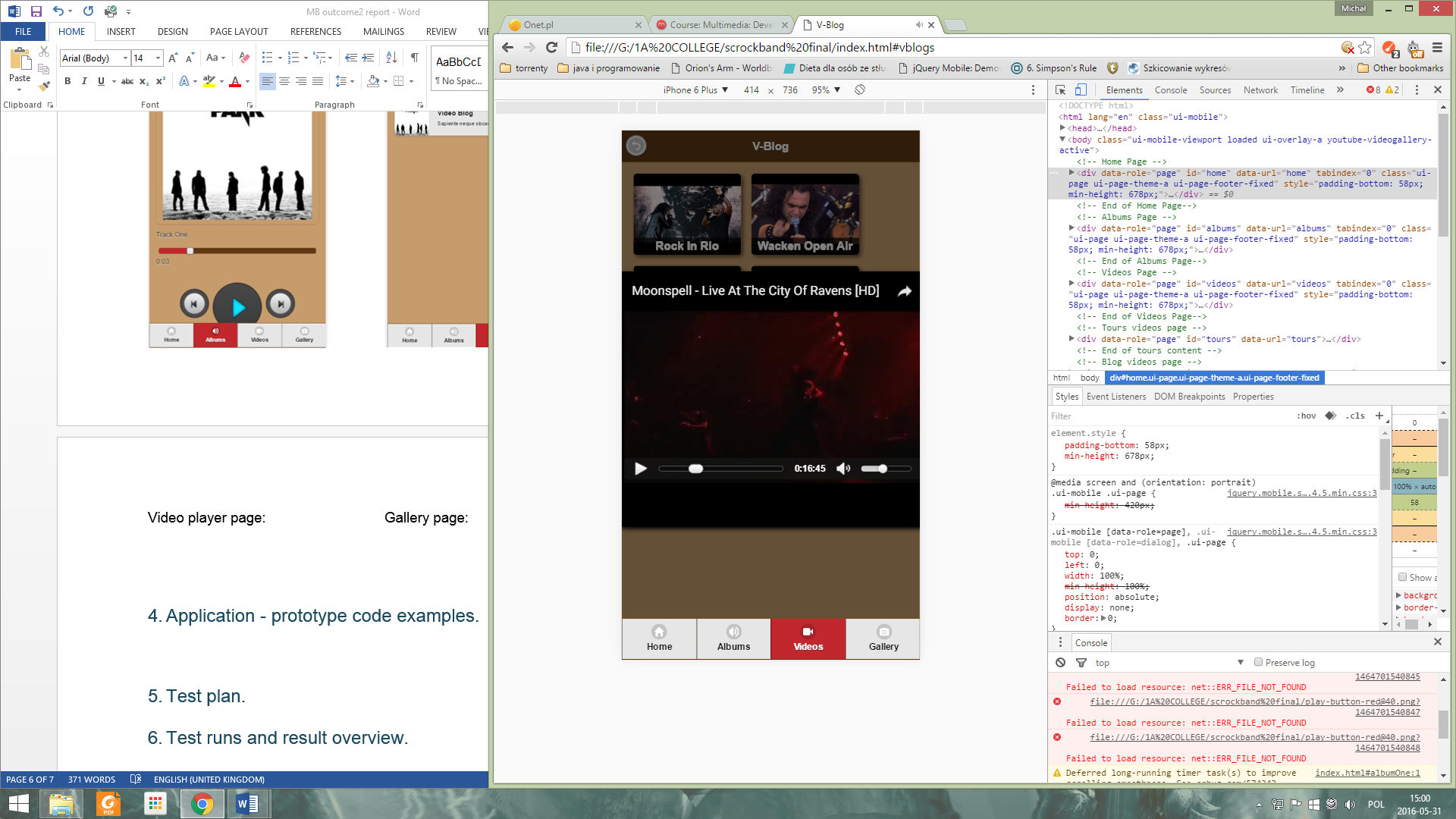
Exit criteria:

* Verify if all tests were run
* Verify if tests covered the area of research valid for the test approach
* Verify the activities were completed on schedule

TEST DELIVERABLES (Output proofs):

Test will produce deliverables in form of screenshots of test outputs for each test run.

  
**Picture 1: Picture 2:**

**Picture 3:** **Picture 4:** **Picture 5:** **Picture 6:**  


TEST ENVIROMENTAL/STAFFING NEEDS:   
  
Test should be performed in closed environment, post-development but before the release to the client (submitting to the College Turnitin), by one or more testers which do not require formal training or advanced programming skills but a thorough knowledge of the requirement specifications is needed.

TEST RESULT OVERVIEW:   
  
All test were passed successfully thus prototype can be released to the client for acceptance, no remediation is required.