# Product owner

A csapatban produt owner feladatkört láttam el, nagy hangsúlyt fektettem a csapat megfelelő összetételének kialakítására, ugyanakkor a tehetségeket figyelembevéve igyekeztem az elvárásokat prezentálni a csapat felé. Úgy gondolom sikerült egy jól működő az esetleges felmerülő problémákra jól reagáló csapatot összekovácsolni. Minden részen segítséget nyújtottam a csapattagoknak, legjobb tudásom szerint. A folyamatok menedzselése mellet az esetleges külső függések miatt igyekeztem a megfelelő helyen feloldani a felmerülő problémákat. Mindemelett az évközi jelentések és leadandó dokumentációk elkészülési határidejét is figyelembevéve prezentáltam a csapat haladását. Hétről hétre jeleztem az esetleges felmerülő hibákat, elvárásokat az ütemezés sikeres tartása érdekében.

# A projekt

A projekt célja egy Legyen ön is milliomos játék publikálása 2 platformon (Android és web), illetve demózás IOS platformon. Regisztráció kezelése, adatok tárolása adatbázisban, érzékeny adatok titkosítva. Cordova keretrendszerrel UI megvalósítása. Backend megvalósítása PHP szerverrel.

A projekt sikere nagyban múlik az alap ötlet megvalósíthatóságán, ugyanakkor nagyban függ a projektbe bekapcsolódó emberek mennyire tudnak egy csapatként küzdeni a cél megvalósításának érdekében.

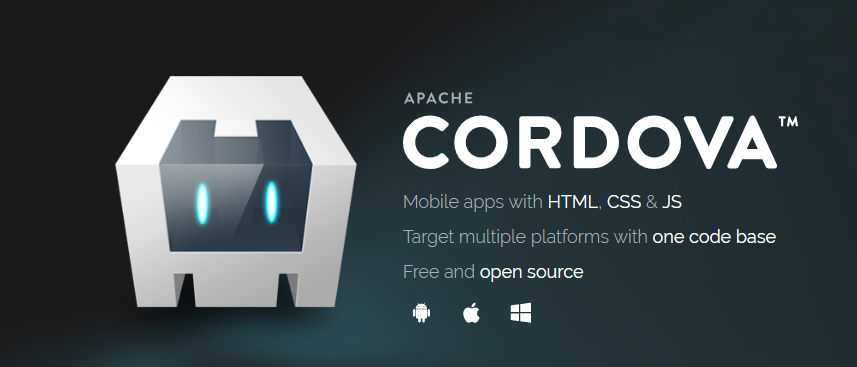
Első lépés ötlet keresés, mely során mérlegelni kell a felmerülő ötlet kivitelezhetőségét, egy eleve rosszul előadott ötletre nehezebb csapatot építeni. A Legyen ön is milliomos önmagában nem egyedi ötlet van már a piacon számtalan verziója sok csapat megvalósította már. Ami ezen tények ellenére is sikeressé tudták tenni az ötletet az a megvalósítási technológia, mely forradalmi és jelen pillanatban kevéssé kiaknázott a fejlesztők körében. Sok fejlesztő környezet és eszköz létezik, de kevés van melynek használatával felhasználhatóvá lefordíthatóvá válik a kód a két legjobban elterjedt mobil operációs rendszerre, ugyanakkor a számítógépet előnyben részesítőket is a kód gyökeres átgondolása nélkül megszólíthat a fejlesztő csapat. Ezen nagy lefedettség tudatában alakult ki a projekt ötlete véglegesen.

Második lépés a csapat kialakítása. Létfontosságú a jó csapat összetétele, a projekt sikerességéhez szükséges a stabil háttér rendszer mivel az applikáció szívét adja, ezen rendszer a kritikus pontja a projekt sikerességének, így elengedhetetlen egy a hálózatokat jól ismerő csapattag. Ugyanakkor egy embernek ezen feladat az adattárólástól a szerver oldali programozáson át nagy feladat így a projekten két ember is megkapta ezen feladatkört. Szükséges a játék dinamikáját és a háttér rendszer közötti részt is jól megvalósítani ezen hiányosságok szintén a teljes projekt vesztét okoznák, így a JS feladatokat szintén két ember kapta meg faladatául. Már csak egy rész nem került említésre ugyanakkor ez is ugyanannyira fontos, mint az előző két rész az applikáció megjelenése mely CSS segítségével lett kialakítva az alap HTML kódból, erre a feladatra egy ember lett megbízva a csapatból. A csapattagok csatlakozása után kezdetét vehette a megvalósítás megtervezése.

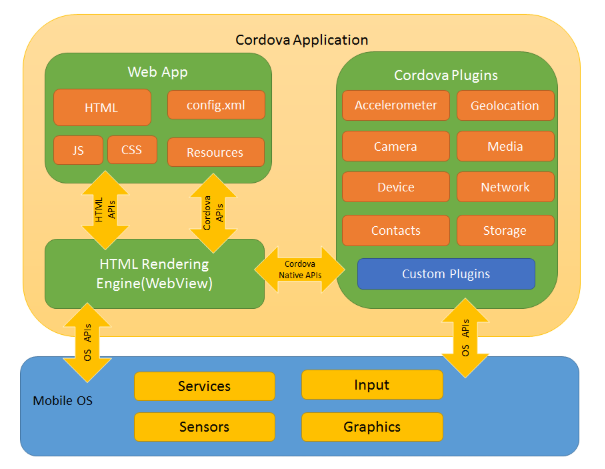
# A tervezés

Egy applikáció fejlesztése nehéz feladat, mivel nehéz döntéseket kell meghozni még a program megírása előtt. Elsősorban fel kellett mérni a csoport preferenciáit. Ez alatt programozási nyelvismeret, integrált fejlesztői környezet (IDE) [[1]](#footnote-1) kiválasztásán volt a legnagyobb hangsúly. Emellett el kellett dönteni, hogy milyen erőforrásra volt szükség. Egy webes / Android applikáció legfőbbként 2 részre osztható: Backend, Frontend. Ezeket a részeket több alrészre is kellett osztani, annak érdekében, hogy megosztott legyen a csapat erőforrása. Személyes megbeszélés során a csapat egyet értett, hogy Cordova keretrendszer lenne a legmegfelelőbb az applikáció fejlesztésére. Emellett az IDE, amit mindenki ismert az a Visual Studio. Egyéb lehetőségek között lehetett volna a Netbeans is. A fejlesztéshez szükséges többi erőforrás: Scrum Master, Product Owner, Frontend Developer, Backend Developer, Frontend Designer.

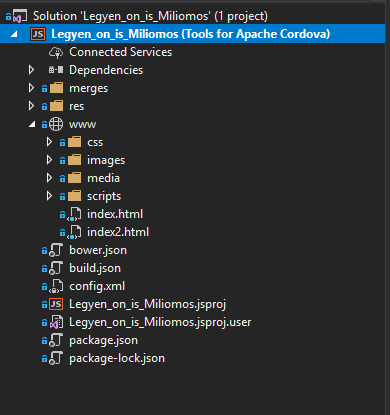
## Cordova



,,Az Apache Cordova egy nyílt forráskódú mobilfejlesztési keret. Ez lehetővé teszi a szabványos webes technológiák - HTML5, CSS3 és JavaScript - használatát a platformok közötti fejlesztéshez. Az alkalmazások az egyes platformokra célzott csomagolásokon belül futnak, és a szabványoknak megfelelő API-kötésekre támaszkodnak az egyes eszközök képességeihez, például érzékelőkhez, adatokhoz, hálózati állapothoz stb.”[[2]](#footnote-2)



1. ábra Cordova felépítése

A Cordova projektet létrehozva az alábbi mappaszerkezet alakul ki

2. ábra Cordova mappa struktúra

A WWW mappa alá kerül be az eszközökre befordítandó mappastruktúra.

A css mappa tartalmazza a kinézetért felelős css fájlokat.

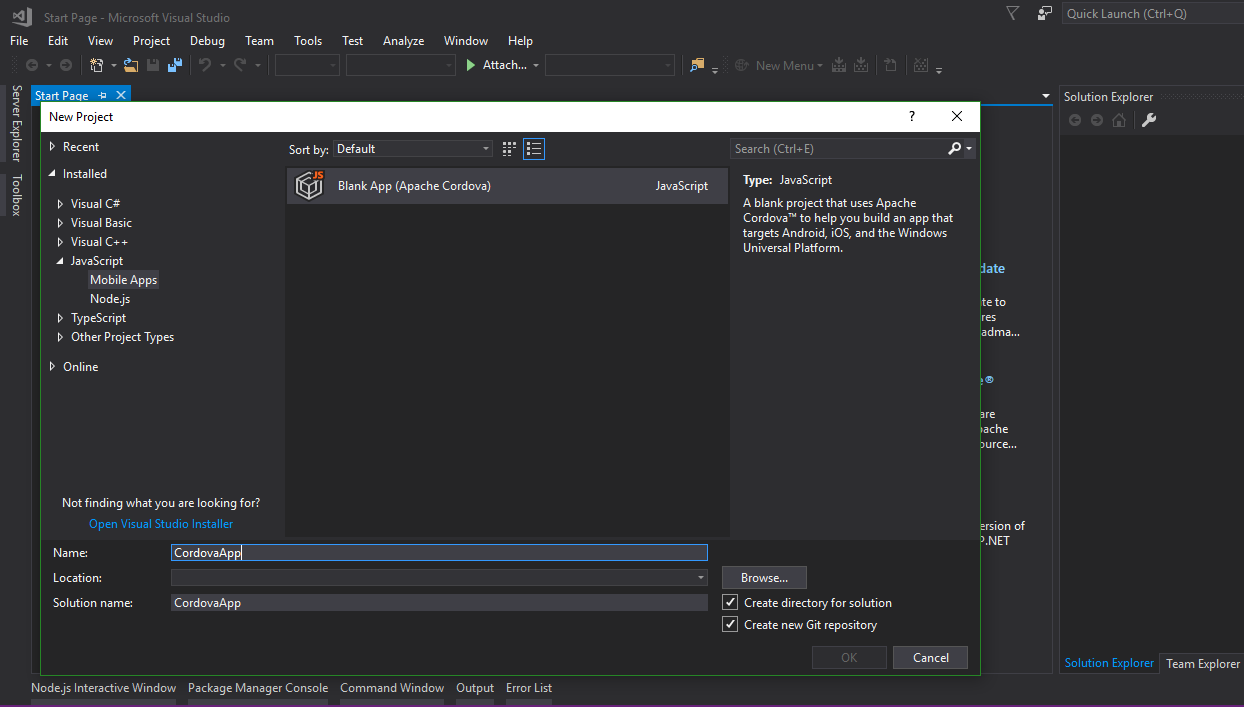
A media mappa az alkalmazásban előforduló képeknek ad helyet.

A script mappa a JS-t tartalmazza mely vezérli az egyes interakciók esetén végrehajtandó feladatokat.

## Visual Stúdió

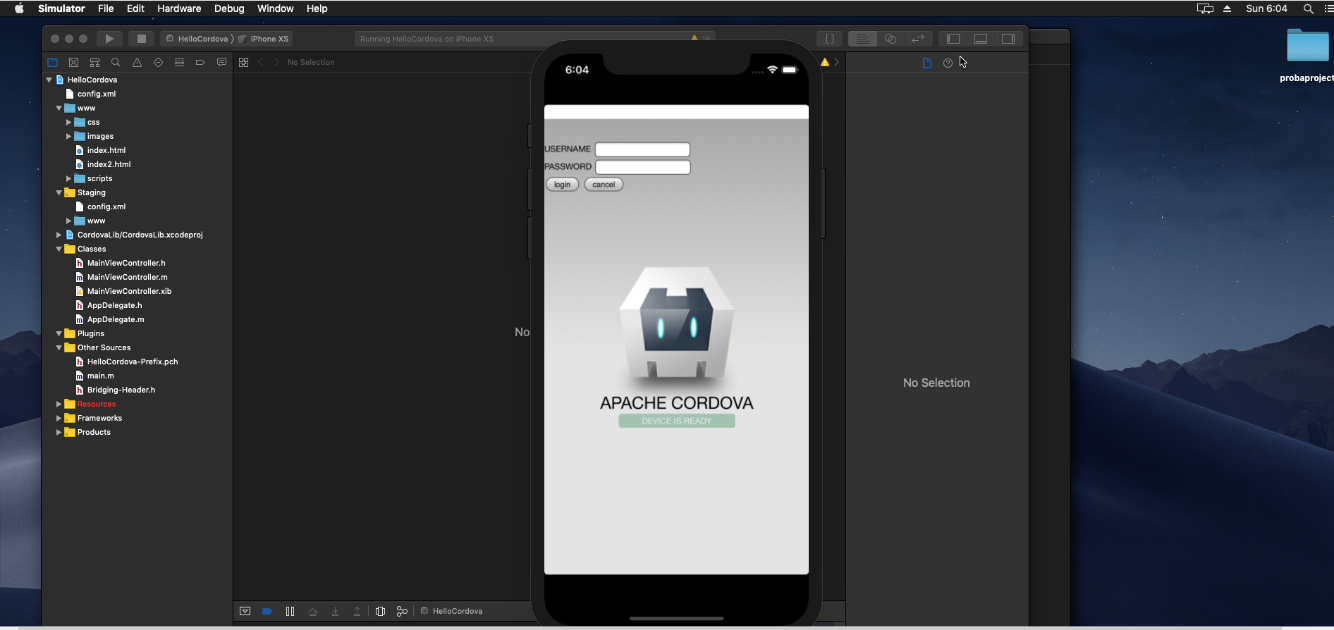
Fejlesztői eszköznek a Visual Stúdió-ra esett a választás, mely támogatja a Cordova projekt létrehozását, elkészít egy alap projektet mely készen is áll a fordításra mindhárom platformra. Ugyanakkor ez egy statikusan megjelenő oldal, de a számunkra fontos már említett mappaszerkezetet legenerálja.

A Visual stúdió segítségével lehet fejleszteni a JS, CSS és HTML típusú fájlokat. Ennek segítségével a fordításra kerülő rész teljes egészében elkészíthető.

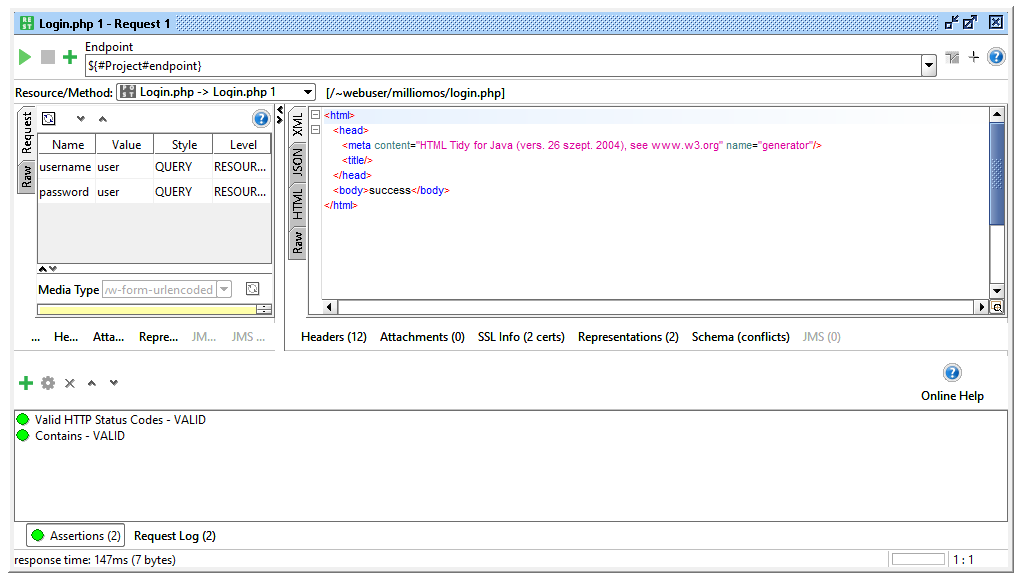


3. ábra A létrehozás folyamata

## XCODE

Ezen program segítségével tudjuk emulálni a Apple telefonra készített IOS rendszerét és ezáltal a program működését megfigyelni, Az Xcode 10 mindent tartalmaz, amire szüksége van ahhoz, hogy minden Apple platformhoz csodálatos alkalmazásokat hozzon létre.”[[3]](#footnote-3)Az Xcode is támogatja a cordova projektet. Ennek köszönhetően a virtuális gépen mindössze az emulálás és fordítás folyamatát kellet elvégezni.

## SoapUI

Ezen program a háttérrendszer tesztelési folyamatát könnyítette meg a csapat számára. Beküldhető vele a szerverre egy kérés melyre a szerver válaszol tudjuk vizsgálni, hogy a kapott válasz megfele-e a várt formátumnak, tartalmilag annyi és olyan információt kapunk-e vissza, amit vártunk. 

4. ábra Sikeres belépés

## A játék, és szabályai

A Legyen ön is milliomos egy televíziós kvízjáték [[4]](#footnote-4). A játék célja, hogy minél több kérdésre válaszoljon a játékos helyesen. A játékos nyereménye annál nagyobb, minél több kérdésre válaszol helyesen jól. A kérdésekre 4 válasz lehetőséget kap a játékos, ezek közül csak egy helyes válasz van. A játékos rendelkezik segítséggel is, amiket csak egyszer használhat egy játék során. A segítség viszont nem garantáltan, de nagy valószínűséggel adnak helyes válasz, ilyenre példa a közönség szavazata.

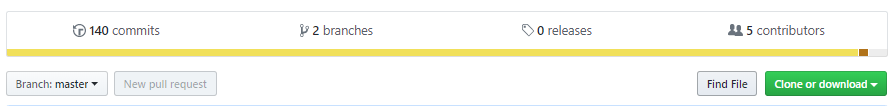
## A projekt

Az applikáció egyéniségét a téma adja, mivel az egyetemmel kapcsolatos kérdéseket ad a játékosnak. Például Számítógépek és Távközlési hálózatok kérdésekre kell válaszolnia a játékosnak, a következő kérdésben pedig már más kurzusról kap kérdést. Az applikáció elsősorban telefonkészüléken fut majd, de internetes böngészőn keresztül is lehet vele játszani. A játékos ugyanazt az élményt kapja, minden platformon.

## Git

A Git egy változást irányító rendszer, az applikáció fejlesztésre ajánlott, mivel nem csak egy ember dolgozik az applikáción. A Git segítségével egy internetes tárolón van a projekt, erről a tárolóról le lehet menteni, és fel lehet tölteni a változtatásainkat, amit program kódján vagy különböző részein hajtottunk végre.

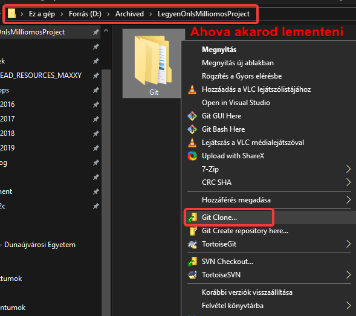
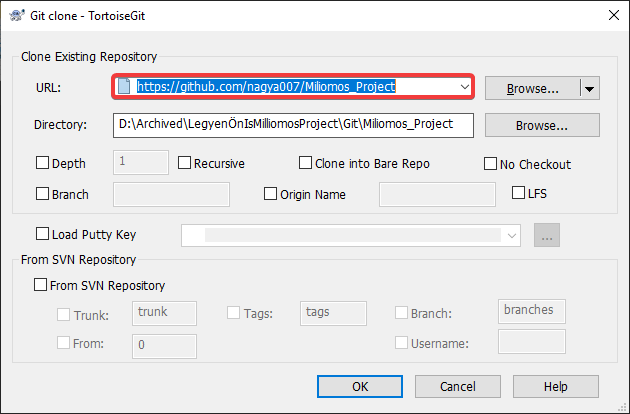
Az irányítás alatt pedig arról beszélünk, hogy minden változást, amit a fejlesztők végre hajtanak, azt menti az internetes tároló. Így lehetőség van nyomon követni változásokat, esetleg visszavonni változásokat.



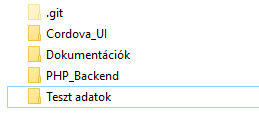
5. ábra A Git követi a változásokat

* 1. Használata a Legyen ön is Milliomos projektben

A Git-et lehet használni a hozzávaló terminállal, vagy *TortoiseGit* [[5]](#footnote-5)használatával is. TortoiseGit-el Windows alapú GUI-t kapunk a Git használatához. Emellett rendelkezni kell egy Git felhasználóval is, aminek van engedélye a projekt letöltéséhez. A projekt tagok az e-mailjük segítségével regisztráltak a Git honlapjára.[[6]](#footnote-6)



A távoli tárolóról letöltött projektet ezután lehet már használni is, ha pedig szeretnénk feltölteni módosítást a tárolóra, akkor a Git push funkcióját kell használni. Ha pedig mások módosításait akarjunk letölteni, akkor a Git pull funkcióját kell használni. Ajánlott letölteni minden módosítást, még mielőtt mi akarunk módosítani rajta. A TortoiseGit lehetővé teszi ennek a folyamatnak a felgyorsítását, mivel egy GUI[[7]](#footnote-7)-t ad a Git szoftvernek. Így elég gomb kattintással fel és letölteni a projekt fájlokat.

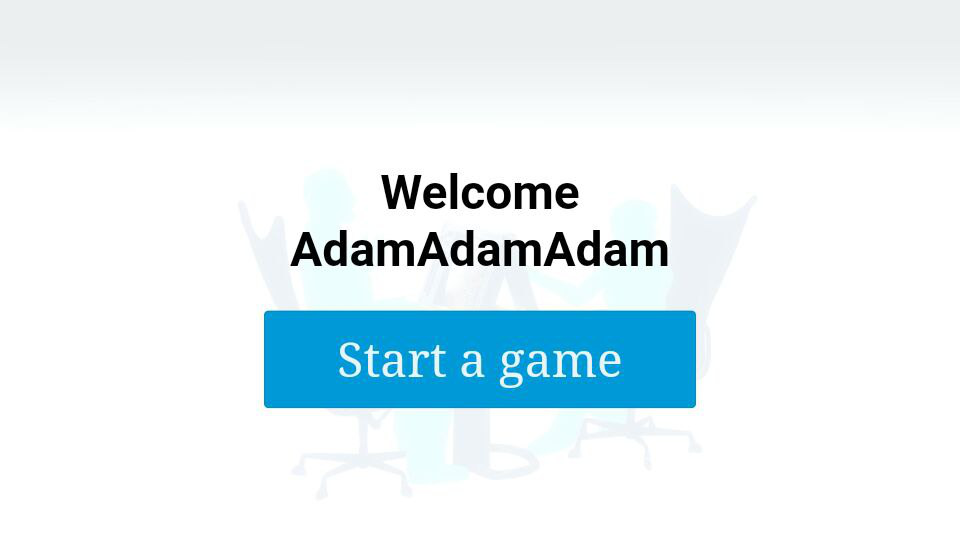
A jól szeparáltság érdekében, és a közös munka megkönnyítése érdekében külön mappaszerkezet került kialakításra a backend és a frontend tárolására, ezek mellet továbbá dokumentációknak szintén egy külön mappa lett kialakítva a verziókezelő rendszerben a teszt szintén egy saját mappát kapott a projekt előre haladásával. 

6. ábra Projekt mappaszerkezet

## Dokumentálások

Projekt folyamata során bővített dokumentumok. A beszélgetések, összejöveteleket egy eseménynaplóba rögzítve lettek, Excel formátumba. A beszélgetések rögzítésével újabb teljesítendő célokat lehetett megszabni, ennek segítségével mindig van egy elérhető dokumentum, ami tartalmazza a teendőket, és korábbi felmerült kérdéseket. A személyes összejövetelek és az online beszélgetés rendkívül segítette az applikáció gyors fejlődését, és a dokumentumok is ezt tükrözik. A dokumentálás része volt a fontosabb hivatkozások mentése, például képek vagy linkek. Az így mentett adatok nem vesznek el soha, mivel a Git verzió kezelés is elősegítette a dokumentumok bővítését.

# Tesztelés a projekt kritériumok szerint

A projekt megbeszélésekre tisztában kellet lenni, azzal, hogy hol tart pontosan a projekt, mi a következő lépés. Ezen feladatot Product Owner látta el. A backend tesztelés a már említett SoapUI programmal történt. Minden php fájl egyenként tesztelhető, szükséges beküldeni a megfelelő adatokat majd az érkező válasz kiértékelhető. Ugyanakkor a hibás adatok esetén is kell visszajelzést kapni melyek szintén tesztelhetőek ezen módszerrel. Elengedhetetlen a JS kód átfutása és fordításának ellenőrzése is, melyet a CSS kóddal egyidőben lehet tesztelni. A telefonos tesztek alatt több hiba kibukott melyeket a heti szinten esedékes megbeszéléseken megvitattunk kitűztük az adott hiba javításának határidejét és felelősét. 

1. https://tudasbazis.sulinet.hu/hu/szakkepzes/informatika/a-program-gyakorlati-megvalositasa/szamitogepes-programozas-a-gyakorlatban/a-vizualis-fejlesztoeszkozok-megismerese-hasznalata [↑](#footnote-ref-1)
2. https://cordova.apache.org/ [↑](#footnote-ref-2)
3. <https://developer.apple.com/xcode/> [↑](#footnote-ref-3)
4. https://hu.wikipedia.org/wiki/Legyen\_%C3%96n\_is\_milliomos! [↑](#footnote-ref-4)
5. https://tortoisegit.org/ [↑](#footnote-ref-5)
6. https://github.com/ [↑](#footnote-ref-6)
7. Graphical user interface [↑](#footnote-ref-7)