

הנדסת תוכנה – תירגול

היכרות עם כלי הפיתוח בקורס



שרת לקוח ואפליקציות רשת

* שרת – נותן שירות. ממתין תמיד ללקוחות.

* לקוח – מבקש השירות

* אפליקציית רשת – מימוש שרת-לקוח מקיף, מרגע הבקשה ועד קבלת התשובה.

MEAN Stack - הקדמה

* בניית web applications כרוכה בשימוש בטכנולוגיות וכלים שונים:
מסד נתונים, לוגיקה בצד השרת, טיפול והצגה של הנתונים בצד הלקוח.

* מהו MEAN?

“MEAN is a fullstack JavaScript platform for modern web applications”

- * **MongoDB** - the database
- * **Express** - the web framework (URL routing)
- * **AngularJS** - as the frontend framework
- * **NodeJS** - as the server platform

Azure - הקדמה

- * Azure הינה תשתית פיתוח בענן של מיקרוסופט
- * פיתוח אפליקציות לכל הפלטפורמות בצורה מהירה וזולה
- * על גבי Azure ניתן לפתח אפליקציות לכל הפלטפורמות בצורה מהירה (Java, PHP, Node.js, Python, Ruby, .NET ונוספות).
- * בקורס, ניתן להשתמש ב- Azure App Service להרצת אפליקציית Web שניצור, אותה נפרסם ב-GitHub (הסבר בהמשך)



Git - הקדמה

- * Git היא מערכת בקרת גרסאות קוד שפותחה במקור עבור פיתוח מערכת ההפעלה לינוקס.
- * בעזרת הכלי הזה אנו יכולים לנהל גרסאות קוד, לעבוד בצוותים, ולאחד מצבים שונים של עבודה.
- * שימוש יעיל של בקרת גרסאות היא מיומנות חשובה ומועילה עבור כל מפתח ובמיוחד אם יש יותר ממפתח אחד שמעורב בפרויקט.
- * בעבר, ואפילו בלימודים, הייתם שומרים קובץ שונה בכל פעם שעשיתם שינוי שלא בטוח שישאר בתוכנה.
- * היום נלמד איך עושים את זה **נכון**

GitHub

- GitHub הוא שרות git אונליין הפופולרי בעולם.
- משמש כמערכת ניהול פרוייקטי תוכנה, רשת חברתית, מקום לקוד פתוח, ועוד.
- בקורס נשתמש ב-github בצורה נרחבת.

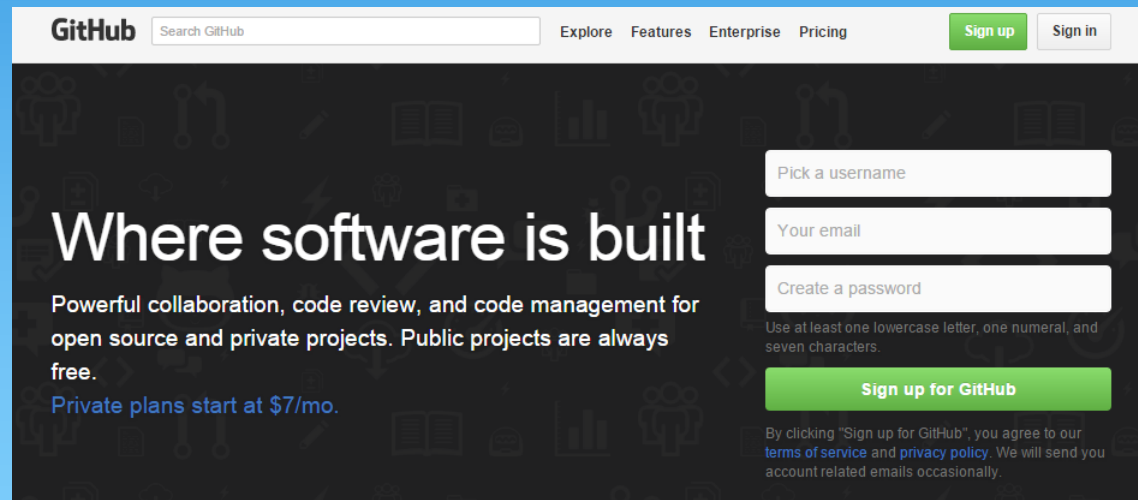
כיצד להשתמש ב-GitHub?

<https://git-scm.com/downloads>

* התקן את git בקישור:
או השתמש ב-command line

<https://github.com>

* צור חשבון ב-Github (חינם) בקישור:




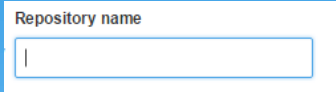
The screenshot shows the GitHub homepage with a dark background. At the top, there's a navigation bar with the GitHub logo, a search bar, and links for Explore, Features, Enterprise, and Pricing. On the right, there are 'Sign up' and 'Sign in' buttons. The main content area features the text 'Where software is built' and a description of GitHub's capabilities. On the right side, there's a sign-up form with fields for 'Pick a username', 'Your email', and 'Create a password'. Below these fields is a green 'Sign up for GitHub' button. At the bottom of the form, there's a small disclaimer about terms of service and privacy policy.



* צור repository (בשקופיות הבאות)

ניתן להתקין את GitHub Desktop ל-MAC או ל-WINDOWS בקישור - <https://desktop.github.com/>

יצירת repository – אפשרות 1:

- * **Log onto github**
- * **Click on  to add repository**
- * **Enter name  and click create repository**
- * **Add .gitignore file**
- * **Having a README.md counts as a good practice**
- * **Click on Settings to control access (Collaborators tab)**
- * **Clone the repository to your PC**

יצירת repository – אפשרות 2:

- * **Create a folder for your project**
- * **Create your local repo by typing:**
 - * **git init** – initialize the repo.
 - * **Add/Create needed files.**
 - * **git commit -m “Initial commit”** – save the state
- * **Create a repo on github**
- * **Add a “shortname” for your git repository**
 - * **git remote add [shortname] [url]**
git remote add origin https://github.com:[user name]/[repository name].git
Remember: You can copy/paste the repo url from github
- * **Push your code onto github**
 - * **git push -u [remote-name] [branch-name].**

workflow

Person A

➤ Setup project & repo

➤ push code onto

github →

➤ edit/commit

➤ edit/commit

➤ pull/push ←



Person B

→ ➤ clone code from github

← ➤ edit/commit/push

➤ edit...

➤ edit... commit

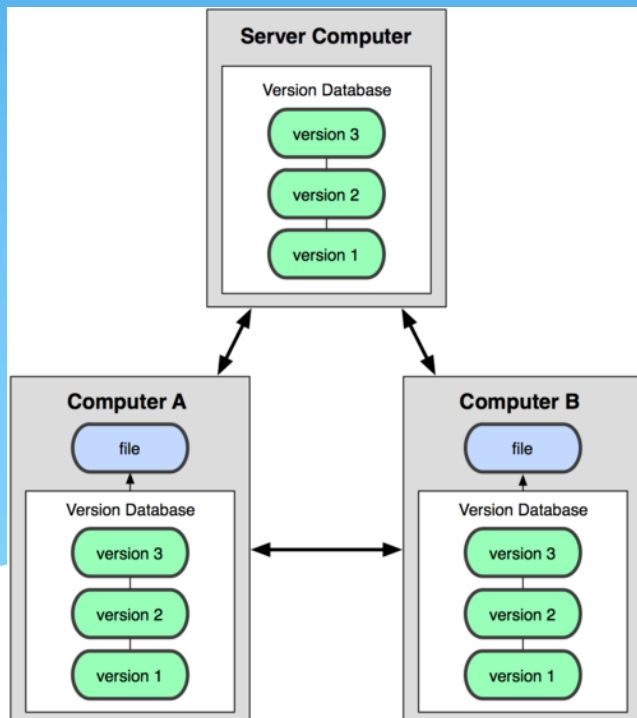
→ ➤ pull/push
←

This is just the flow, specific commands on following slides.

It's also possible to create your project first on github, then clone (i.e., no git init)

Collaborating via github - cloning

- * **git clone** adds the remote repository under the name **origin**
- * **git clone https://github.com:[user name]/[repository name].git**



Git commands

command	description
git clone url [dir]	copy a git repository so you can add to it
git add files	adds file contents to the staging area
git commit	records a snapshot of the staging area
git status	view the status of your files in the working directory and staging area
git diff	shows diff of what is staged and what is modified but unstaged
git help [command]	get help info about a particular command
git pull	fetch from a remote repo and try to merge into the current branch
git push	push your new branches and data to a remote repository
others: init, reset, branch, checkout, merge, log, tag	

GitHub Wiki - הקדמה

* כשם שכתובת קוד ובדיקות חשובים, תיעוד מצוין עוזר לאחרים להשתמש ולהרחיב את פרויקט.



* כל מאגר GitHub מגיע עם חלק עבור תיעוד, שנקרא Wiki.

* GitHub Wiki הוא מקום במאגר, בו ניתן לשתף תוכן רב על הפרויקט כגון: איך להשתמש בו, איך זה תוכנן וכן הלאה.

* ניתן לערוך ישירות ב-GitHub או בעזרת עורך טקסט פשוט ולהעלות אח"כ את השינויים.

* ניתן להגדיר הרשאות עריכה במאגר למשתמשים שונים.

* בקורס, ניצור Wiki עבור הפרויקט.









דוגמא - GitHub Wiki

Create new page

Home

Write Preview

h1 h2 h3   B i <>     ? Edit mode: Markdown

Block Elements

Span Elements

Miscellaneous

Paragraphs & Breaks

Headers

Blockquotes

Lists

Code Blocks

Horizontal Rules

To create a paragraph, simply create a block of text that is not separated by one or more blank lines. Blocks of text separated by one or more blank lines will be parsed as paragraphs.

If you want to create a line break, and a line with two or more spaces, then hit Return/Enter.

Welcome to the Test wiki!

Edit Message

Initial Home page

Save Page

* להוספת עמוד חדש ב-Wiki

Create the first page

לחץ על

* השתמש בעורך הטקסט כדי

להוסיף את תוכן הדף שלך.

* הזן הודעה המתארת את הדף

החדש שאתה מוסיף.

* כדי לאשר את השינויים לחץ

Save Page

על

* ניתן לערוך גם באמצעות GitHub Desktop



פורום/צ'אט וקבוצת הקורס

* לאחר פתיחת חשבון ב-github, נדרש להתחבר ל:

* צ'אט/ פורום הקורס, בקישור:

* <https://gitter.im/jce-il/se-class/2016b>

* קבוצת הקורס Students2016bM, בקישור:

* <https://github.com/orgs/jce-il/teams/students2016b>



משימה אישית 1

* משימה אישית 1:

* <http://jce-il.github.io/se-class16b/morea/webapp/experience-webapp-deploy-starter.html>

* הגשת משימה אישית 1:

* <http://jce-il.github.io/se-class16b/assessments/#assessment-personal1-deploy-webapp>

* תאריך הגשה סופי ולו"ז:

* 8/3/2016

* <http://jce-il.github.io/se-class16b/schedule/>

משימה אישית 1 – שלבים:



- * יצירת חשבון ב-github
- * חיבור לצ'אט ולקבוצת הקורס
- * Fork למאגר הקיים ב-github
- * בקשת קוד ל-azure בצ'אט
- * יצירת חשבון ב-azure (ניתן להיכנס עם המייל של המכללה)
- * יצירת אפליקציה ב-azure
- * חיבור המאגר ב-github לאפליקציה ב-azure
- * הרצת האפליקציה (בדיקה)
- * שינוי כותרת בדף הראשי ('Allo, 'Allo!) – ב-github (online/offline)
- * Sync ב-azure והרצת האפליקציה (בדיקה)
- * הגשת טופס משימה אישית 1
- * כיבוי האפליקציה ב-azure

בהצלחה!

Git Resources

- * **Free on-line book:** <http://git-scm.com/book>
- * **Git tutorial:** <http://schacon.github.com/git/gittutorial.html>
- * **Reference page for Git:** <http://gitref.org/index.html>
- * **Git website:** <http://git-scm.com/>