**המחלקה להנדסת תוכנה**

**פרויקט גמר – תשע"ח**

**אתר אינטרנט לניהול מרפאת שיניים**

**Web site for managing a dental office**

**מאת**

**דוד קלינר**

**מנחה אקדמי: מר רוג'ר כהן אישור: תאריך:**

**אחראי תעשייתי: פרופ'/דר'/גב'/מר ........ אישור: תאריך:**

**רכז הפרויקטים: פרופ'/דר' ........ אישור: תאריך:**

מערכות ניהול הפרויקט: Bitbucket

|  |  |  |
| --- | --- | --- |
| # | מערכת | מיקום |
| 1 | מאגר קוד | https://github.com/davidkaliner/Dental-Office |
| 2 | יומן | https://bitbucket.org/davidkaliner/web-site-for-managing-a-dental-office/addon/bitbucket-trello-addon/trello-board |
| 3 | ניהול פרויקט (אם בשימוש) | https://bitbucket.org/davidkaliner/web-site-for-managing-a-dental-office |
| 4 | הפצה |  |
| 5 | סרטון גירסת אלפא | <https://www.youtube.com/watch?v=bs-Gj-3S56g&feature=youtu.be> |

תוכן עניינים:

[1. מבוא 5](#_Toc504920295)

[2. תיאור הבעיה 6](#_Toc504920296)

[דרישות ואפיון הבעיה 6](#_Toc504920297)

[הבעיה מבחינת הנדסת תוכנה 7](#_Toc504920298)

[3. תיאור הפתרון 7](#_Toc504920299)

[מהי המערכת 8](#_Toc504920300)

[תהליכים ונתוני המערכת 8](#_Toc504920301)

[תיאור הפתרון המוצע 9](#_Toc504920302)

[תיאור הכלים המשמשים לפתרון 9](#_Toc504920303)

[4. תכנית הבדיקות 9](#_Toc504920304)

[5. סקירת עבודות דומות בספרות והשוואה 9](#_Toc504920305)

[6. נספחים 11](#_Toc504920306)

[א. רשימת ספרות \ ביבליוגרפיה 11](#_Toc504920307)

[ב. תרשימים וטבלאות 11](#_Toc504920308)

[ג. תכנון הפרויקט 13](#_Toc504920309)

[ד. טבלת סיכונים 14](#_Toc504920310)

[ה. רשימת\טבלת דרישות 15](#_Toc504920311)

תקציר הפרויקט:

ישנם בשוק היום מבחר רב של תוכנות לניהול מרפאות רפואיות בכלל ומרפאות שיניים בפרט. רוב האופציות הקיימות באים לתת מענה לכלל צרכי המערכת הרפואית, דבר שלא הכרחי למרפאות קטנות. הדרישה לתת מענה כמה שיותר רחב לכלל הצרכים של המרפאות גורם לתוכנות אלה להיות מאוד מסורבלים, מסובכים ויקרים. בכוונתי לעצב מערכת מינימליסטית ככל הניתן אשר תכלול רק את הפונקציות הרלוונטיות למרפאה קטנה וגם תאפשר תקשורת עם המטופלים.

מונחים:

* מרפאה קטנה : מרפאה בא עובדים מספר קטן של רופאים (אחד או שניים לרוב).
* תוכנות ניהול: כלל הפתרונות הקיימים בשוק היום לניהול מרפאות.
* רופא\איש צוות : בעל הרשאות במערכת
* מטופל: לקוח המשתמש במערכת

# מבוא

רקע כללי: ישנם כמה תוכנות כרגע בשימוש לניהול רפואי בכללי ומרפאות שיניים בפרט אך כולם נותנים מענה רק לצד הצוות הרפואי ולא ללקוחות. בכוונתי לעצב מערכת "קלה" יותר אשר תיתן מענה גם לצרכי הלקוחות (המטופלים) וגם לרופאים. כוונתי במילה "קלה" היא שלא יהיה כלול במערכת כל נגיעה כספית – הפקת חשבונית וכו'. כמו כן אדאג שהממשק משתמש יהיה קל ונוח לתפעול (בניגוד לתוכנות המסובכות יותר).

היות והאתר ישמש גם לרופאי השיניים וגם ללקוחות, את ההפרדה נעשה ע"י Login שונה. האתר (מצד הלקוחות) יכלול את האלמנטים הבאים:

1. הסבר כללי על כל טיפול כולל מחיר ומשך זמן הטיפול.
2. הסבר אודות הצוות הרפואי
3. מערכת להזמנת תורים (בדומה למערכת בקולנוע)
4. מאמרים המכילים מידע חשוב ושימושי
5. מעקב על מצב טיפול- מה בוצע מה יבוצע ומצב תשלום

בכוונתי שמערכת הזמנת התורים תהיה בגדר בקשה – כלומר יידרש אישור של הרופא על מנת לשנות תור וזאת עקב אילוצי זמן. יש טיפולים מסובכים שלוקחים יותר זמן וייתכן שמה שביקש המטופל לא מתאים.

את פרטי המטופלים והרופאים יישמרו בdatabase – כלומר יישמר המידע עבור כל ישות (רופא או מטופל) את המידע הרלוונטי:

-פרטים אישיים: שם, כתובת , טלפון

-סטטוס (מטופל או צוות רפואי)

-תוכנית טיפול

-טיפול אחרון שבוצע

-תורים עתידיים

-מצב כספי

כל אלה ייכנסו לdatabase וישמשו לאתר.

צד מנהל (צוות רפואי): תהיה להם אפשרות לאשר בקשת שינוי תור או לא לאשר. לקבוע תוכנית טיפול עבור כל מטופל ולסמן כל שלב בתוכנית טיפול כבוצע או לא כמו כן לשנות תוכנית טיפול קיימת. תהיה לצוות אפשרות לערוך ולשנות נתונים של הסברי טיפול ומחירים. תהיה להם אפשרות להעלות ולהוריד מהאתר מאמרים רפואיים.

מערכת הזמנת תורים: בכוונתי לעצב מערכת שבו המטופל יכול לבקש לשנות את התור שנקבע לו. המטופל יראה לוח שנה שבו יבחר יום. לאחר בחירת יום יראה לוח שעתי שבו יוכל לבקש שעת טיפול רצויה. לכל טיפול מוגדר זמן שנדרש לאותו טיפול ואם זמן הטיפול הבא לא נכנס בזמן הפנוי לא תתאפשר קביעת התור. לאחר הבקשה של המטופל יקבל המטופל הודעת טקסת למסך שהבקשה עברה לאישור הרופא.

# תיאור הבעיה

רוב תוכנות הניהול הנפוצות כרגע היום בשוק (PrimeSuite , ECLIPSE , Next Services , eClinicalWorks PM ) מיועדים לתת מענה לכמעט כל ההיבטים של ניהול מרפאות אבל *רק* מצד הרופאים. הם מאפשרים הרבה דברים מעבר לניהול בסיסי שהם לאו דווקא מתאמים או שימושיים למרפאה קטנה יחסית. למרפאה בודדה אין צורך במערכת שתתאם זימון תורים בין שני מרפאות נפרדות ואין כל צורך במערכת לקביעת משמרות רופאים אם ישנם רק רופא אחד או שניים במרפאה. דוגמה נוספת- אפשרות לניהול מלאי בתוכנה אינה רלוונטית למרפאה קטנה.

התוצאה הסופית היא שרופאים משלמים על תוכנה ובכלל לא מנצלים את מרבית הפיצ'רים שלו. אלמנט נוסף שחסר בתוכנות אלה היא ממשק מול הלקוחות- כלומר דרך תקשורת איתם (לא טלפונית שכן זה גוזל המון זמן) שתאפשר להם לבקש לשנות תור או לבטל הגעה מבלי לשוחח ישירות עם הרופא.

אסכם בזאת: רוב תוכנות הניהול הקיימות היום מיועדות לתת מענה לארגונים גדולים יחסית וכוללים אלמנטים שאינם שימושיים לארגונים קטנים. בכוונתי לעצב מערכת שתספק מענה למרפאות קטנות ולא יכלול אלמנטים לא שימושיים.

בעיה נוספת היא חוסר התקשורת בין הצוות למטופלים- כיום יש צורך בשיחת טלפון לקביעת תור. כלומר הפעולה מתבצעת ידנית, פעולה שעולה בזמן יקר של הצוות.

מהסתכלות בתוכנות הנפוצות בשוק כיום ניתן לראות את המסובכות של התוכנה. כלומר ריבוי אייקונים וסרגלים, ריבוי אפשריות בחירה וכדומה. מצב זה מאוד מרתיע לקהל אשר לא מתמצה בעולם המחשבים ודורש השקעה של זמן רב יחסית על מנת ללמוד להשתמש בתוכנה. המערכת המוצעת כאן באה לתת מענה בעיקר לתופעה זו- כלומר לספק ממשק קל ומצומצם. הקווים המנחים בבחירת פונקציונליות הם:

1. אם אין צורך בפונקציונליות מסויימת – היא לא תופיע כלל
2. נתמוך בדרך אחת בלבד לעשיית כל פעולה וזאת על מנת להקל על למידת התוכנה
3. אייקונים יקבלו שמות ברורים אשר יצביעו בצורה חד משמעית על תפקידם

כמובן שהתפיסה הזאת גורמת ללא מעט בעיות אפשריות. הימנעות מ"הגדלת ראש" מונעת שינויים קלים בהמשך ומגדילה בצורה משמעותית את הצורך בניהול קשר רציף עם הלקוח על מנת לתאם את התוכנה לצרכי הלקוח.

## דרישות ואפיון הבעיה

ישנם שני משתמשים עיקריים לתוכנה: הרופאים והמטופלים.

המטופלים – יהיה עליהם לעבור תהליך של התחברות (LOGIN). למשתמשים חדשים תהיה כפתור של רישום לאתר. לאחר ההתחברות תוצג להם הפורטל הראשי של המרפאה, כאשר ברקע הלוגו הייחודי לכל מרפאה. יהיו בחלקו העליון של המסך מספר לשוניות אשר כל אחד תקשר את המטופל לדף אחר. יהיה דף המכיל מידע כללי על הטיפולים השונים הניתנים במרפאה. לשונית אחרת תכיל דף קשר עם מספרי טלפון כתובות אי-מייל. תהיה למטופל גישה לתוכנית הטיפולים שלו וכן יוצג לו מצב התשלומים שלו. לשונית אחרת תקשר את המטופל למערכת זימון תורים. יוצג בפניו לוח שנה חודשית בו מסומנים התורים העתידים. תהיה לו אפשרות ע"י לחיצת עכבר או נגיעה בטלפון לבקש להזיז או לבטל תור.

הרופאים יראו מסכים דומים למטופלים רק שלהם יתווסף אפשרות לערוך את המידע בכל לשונית (לשנות את המידע על טיפולים , עלויות וכו")

כל בקשה לשינוי תור של המטופלים יקבלו הרופאים כהודעות טקסת ויישמרו כל ההודעות בתור עד לסיום טיפול של הרופא ויהיה עליהם לאשר או לא לאשר לפי שיקול דעתם. לרופאים תהיה אפשרות לערוך פרטים של המטופלים כלומר לשנות תוכניות טיפול ואפילו למחוק מבסיס הנתונים.

כל משתמש יקבל גישה רק לפרטים האישיים שלו- כלומר יש צורך חד משמעי לשמור על עקרונות של אבטחת מידע. נרחיב: למטופל יהיה סיסמה שהוא יבחר ורק על ידי הסיסמה תתאפשר כניסה למערכת (שכן שמורים במערכת מידע רפואי אישי). בסופו של תהליך לכל מטופל תהיה "זהות" במערכת שיכלול את השם, פרטי תקשורת , מצב כלכלי ותוכנית הטיפול שלו.

הרופאים לעומת זאת חשופים לכלל הפרטים של המשתמשים ורשאים לערוך שינויים בקטגוריות הנ"ל.

## הבעיה מבחינת הנדסת תוכנה

המערכת נמצאת בשימוש משותף של שני גורמים שונים- הצוות והמטופלים. את עבודת האדמיניסטרציה על התוכנה יעשה הרופא, כלומר לו תינתן האפשרות להוסיף , לשנות ולמחוק פריטים מה database . יש צורך להפריד ביניהם על מנת שלא לאפשר גישה למי שאינו רשאי לכך. לשני השחקנים צריך לאפשר גישה לבסיס הנתונים. המטופלים על מנת לבצע רישום והצוות לצרכי עבודה. צריך לוודא שלא תהיה דריסה של נתונים של שחקן אחד על ידי שחקן אחר. אלמנט אחר זה האפשרות להתחבר עם טלפון נייד. על מנת שלא אצטרך לטפל בנפרד באנדרואיד ואפל אתכנן את המערכת כWEB לגמרי. אדאג למצב תאימות למכשירים ניידים.

על מנת שהמערכת תעבוד בצורה מיטבית – יש צורך כמובן בתקשורת רציף בין כל גורמי המערכת: השרת , צד לקוח והdatabase. אלמנט נוסף שבעייתי זה נושא אבטחת מידע. יש צורך ברור בשמירת הפרטיות של משתמשי המערכת ומניעת חדירות למערכת. לא אפשרי להסתפק בקביעת סיסמה בלבד אלא צריך לתת מענה גם לכל נושא sql injecting על צורותיו.

# תיאור הפתרון

ראשית יהיה על המשתמש לבצע תהליך התחברות לאתר, ובהתאם לשם משתמש שיוזן למערכת תוצג בפניו האתר.

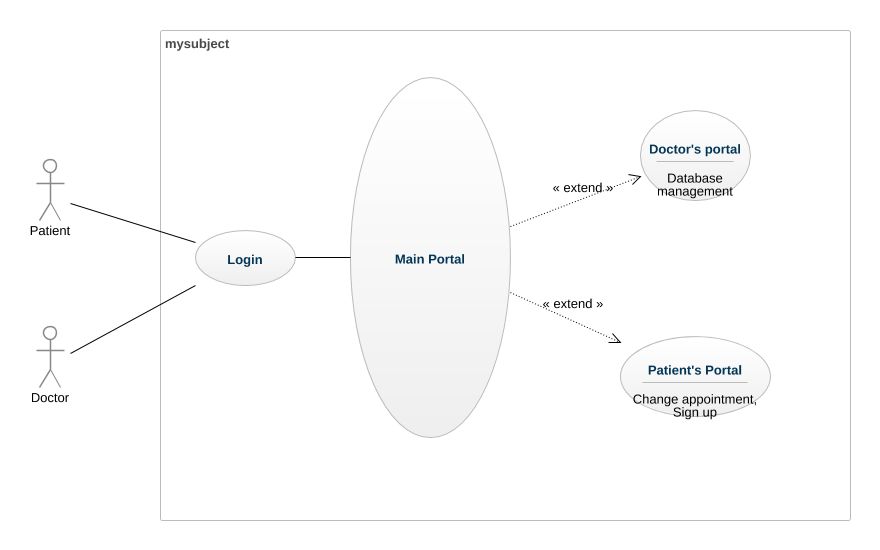
המערכת תציג בפני הלקוח בחלקו העליון מספר לשוניות אשר לכל אחד תחום נפרד. יוצגו מאמרים רפואיים (ולרופא תהיה אפשרות להעלות , לשנות או למחוק אותם) , פרטי התקשרות של הרופאים , הסברים על כל טיפול (שלצוות הרפואי תהיה אפשרות לשנות) ולשונית של זימון תורים. לעיני הלקוח יוצג לוח שנה שבו מסומנים התורים שלו ותהיה לו האפשרות לבטל או לבקש להזיז את התור. הרופאים יראו בלוח שנה את הזמנים התפוסים ויקבלו הודעה מערכת לגבי ביטולים או בקשות שינוי (שאותם הם יצטרכו לאשר). בעת מסירה- יבוצע רישום "בכוח" לרופאים (לפחות לאחד מהם). זה יאפשר מצב בטוח בו רק משתמש בטוח ( הרופא) הוא היחיד שיכול להעניק סמכויות של רופא למשתמש אחר. אם יימסר המערכת ריק- לא תתאפשר ההפרדה.

מודולים מרכזיים בתוכנה:

* מודול שינויי תורים: במודול זה יתנהל מערכת אשר תאפשר למטופל לבקש לשנות את תזמון התור שנקבע לו או לבטל את התור לגמרי. נציין שלא מדובר בשינוי אלא בבקשת שינוי. זאת משום שחלק מהטיפולים הרפואיים לוקחים יותר זמן ויש להשאיר את ההחלטה לשיקול דעת הרופא
* מודול מעקב: מודול העוקב אחרי התקדמות המטופל מבחינת תשלומים ותוכנית טיפול ומציגה לבקשת המטופל או הרופא את מצבו של המטופל בנושאים אלה.
* מודול מידע: מכיל הסברים כללים יחסית על טיפולים המוצעים ללא מחירים שכן כל נושא המחירים זה לפי שיקול דעת הרופאים. גם כלול במודול זה מאמרים בנושאים שונים הקשורים לבריאות השיניים אשר יוצגו בפני המטופלים באתר של המרפאה.

## מהי המערכת

בתרשים ניתן לראות את הארכיטקטורה הכללית של המערכת. שני המשתמשים מבצעים LOGIN למערכת ומקבלים גישה לפי התפקידים לבצע שינויים בDATABASE ולהפעיל פונקציות.



## תהליכים ונתוני המערכת

שני התהליכים המרכזיים של המערכת זה ניהול בסיס הנתונים ומערכת שינוי תזמונים. אף על פי שלשני השחקנים יש גישה לבסיס הנתונים, הגישה של המטופלים בהכרח מוגבל ובחלק מהמקרים כפוף לאישור הרופא.

## תיאור הפתרון המוצע

עבור כל שחקן ששמור בבסיס הנתונים תיווצר שדה של "קבוצה". שדה זה תפריד בין המטופלים לבין הרופאים. לכל שחקן תינתן הרשאות גישה בהתאם לתפקיד. בכל מקרה של חוסר תיאום בין שני השחקנים, הרופא יקבע.

## תיאור הכלים המשמשים לפתרון

אני אשתמש עבור החלק של הקוד הכתוב בJavaScript בסביבת עבודה Brackets , את התרשימים אעשה בGenMyModel . כמובן אשתמש בPHP לבסיס הנתונים.

# תכנית הבדיקות

הבדיקה הראשונה שיבוצע זה כמובן smoke test כלומר בדיקה ראשונית האם המערכת בכלל עובדת. כלומר האם אפשר להיכנס ולהציג אותו על המסך בכלל. לאחר מכן יבוצעו בדיקות תקשורת ראשוניות, כלומר תקשורת עם השרת. נבדוק כתיבת מידע לשרת, שליפת מידע מהשרת ומחיקת מידע מהשרת.

אחרי שיובטח תקשורת תקינה ייבדק נושא אבטחת מידע- כלומר מניעת גישה למי שאינו רשאי והגבלת סמכויות על פי סיווג המשתמש. יבדק גם נושא קלט מהמשתמש. אחרי זה נעבור לuser acceptance testing כלומר יועבר ללקוח לקבלת משוב ושיפורים. יהיו כמה וכמה סבבים של UAT עד לקבלת שביאת רצון מהלקוח.

פירוט תוכנית הבדיקות:

* בדיקת עשן- כלומר בדיקה פשוטה של תקינות המוצרף, דהיינו גישה לאתר
* בדיקות LOGIN: ייבק הימנעות כניסה למי שאינו רשום או לא בעל שם משתמש וסיסמה נכונים
* בדיקות LOGIN : ייבדק הפרדת סמכויות בין מטופלים לרופאים
* בדיקות DATABASE : ייבדק יכולת גישה, קריאה ושינוי נתונים השמורים בבסיס הנתינים
* בדיקות אבטחת מידע: ייבדק נושא אבטחת מידע
* בדיקות UAT : המוצר יעבור שימוש מוגבל אצל הלקוח ויופקו לקחים מהשימוש

# סקירת עבודות דומות בספרות והשוואה

ישנם כמה וכמה תוכנות לניהול מרפאות. המובילים בדירוג כרגע הם PrimeSuite , ECLIPSE , Next Services , eClinicalWorks PM ו Rapid Med . ללא יוצא מן הכלל מדובר בתוכנות גדולות ומורכבות שמטרתם לספק מענה כולל לכלל צרכי המרפאה ולכן כוללים אלמנטים שלא בהכרח מתאימים למרפאה בודדת עם צוות מצוצמם. דוגמה טובה לזה- מערכת ניהול משמרות של רופאים, כאשר מובן שזה מיותר עבור צוות רפואי של 1-2 אנשים. כמו כן אין כל צורך במערכת ניהול תורים בין שתי מרפאות שונות של אותו עסק (מדובר במרפאה עצמאית). בנוסף קיים אפשרות לניהול מלאי- כלומר הזנה של פריטים וכמויות ומעקב של התוכנה על הימצאות פריט. לרוב מטלת ההזנה של שם פריט וכמות נופל על מנהל המרפאה- ולרוב מרפאות קטנות יחסית מוותרות על תפקיד זה.

במה שונה התוכנית המוצעת מהקיימים בשוק?? במילה אחת – פשטות. אם אין צורך בפונקציונליות מסוימת עבור קהל היעד אז היא לא תופיע. משמעו של דבר שמדובר בקהל יעד מסויים וברור שהתוכנה לא מתאימה לכולם. אפשר לטעון שדווקא שוני זה הוא לטובת המשתמשים שכן מדובר במערכת שתוכננה מראש עבור הדרישות שלהם . בנוסף האלמנט האינטרנטי של המוצר מאפשר למרפאה להחזיק אתר שיווקי שגם משמש לניהול לקוחות, מה שוודאי יביא לחיסכון משמעותי של כסף.

מיהו קהל היעד של המערכת? מרפאות עצמאיות, בעלי צוות קטן יחסית ולרוב מבוגרים יותר. קיימת תופעה בה דורות וותיקים יותר (לרוב בני 55 ומעלה) אשר לא גדלו על טכנולוגיית המחשבים נוטים שלא לסמוך על מחשבים יותר מדי. הם מעדיפים דברים פשוטים וקלים להבנה ולשימוש. בזאת מודגשת השינוי בין המערכת המוצעת לקיימים בשוק: מיעוט אייקונים ומינימום פונקציונליות. כמו כן היות והתוכנה מבוססת WEB ולא מותקנת על גבי המחשב (שזו כמובן שוני רציני ) אפשר לנצל את ההיכרות היחסית מעמיקה של קהל היעד עם האינטרנט.

לסיכום: המוצר מבוסס WEB בשונה מכל השאר וזאת על מנת גם לנצל היכרות של כולם עם דפדפני אינטרנט וגם לספק פורטל למטופלים. כמו כן אפשר לעצב את הפורטל של הלקוחות להיות אתר אינטרנט שיווקי לכל דבר עבור המרפאה- שוב בשונה מכל השאר שמתמקדים רק בניהול. המוצר מיועד להיות אינטואיטיבית וקלה לשימוש ככל שניתן. תחום נוסף ייחודי למערכת המוצעת הוא הממשק מול הלקוח להזיז או לבטל תור. זה מתאפשר בגלל האופי הWEBי של המוצר. תתאפשר גם גישה מטלפונים ניידים (שכן הרבה יותר נפוצים בשימוש יום יומי). השוני העיקרי הוא ברעיון המרכזי שלפיו עיצבו את המערכת. המערכות הקיימות תוכנו מראש לתת מענה כולל לכל הצרכים הפנימיים של המרפאה. כלומר תקשורת מול לקוחות אינו חלק מהמערכת. כתוצאה מזה בהכרח על הצוות לנהל בעצמן את התקשרות מול הלקוחות.

# נספחים

ספרות, תרשימים נוספים, תכנון הפרויקט, טבלת ניהול סיכונים, טבלת דרישות (URD),

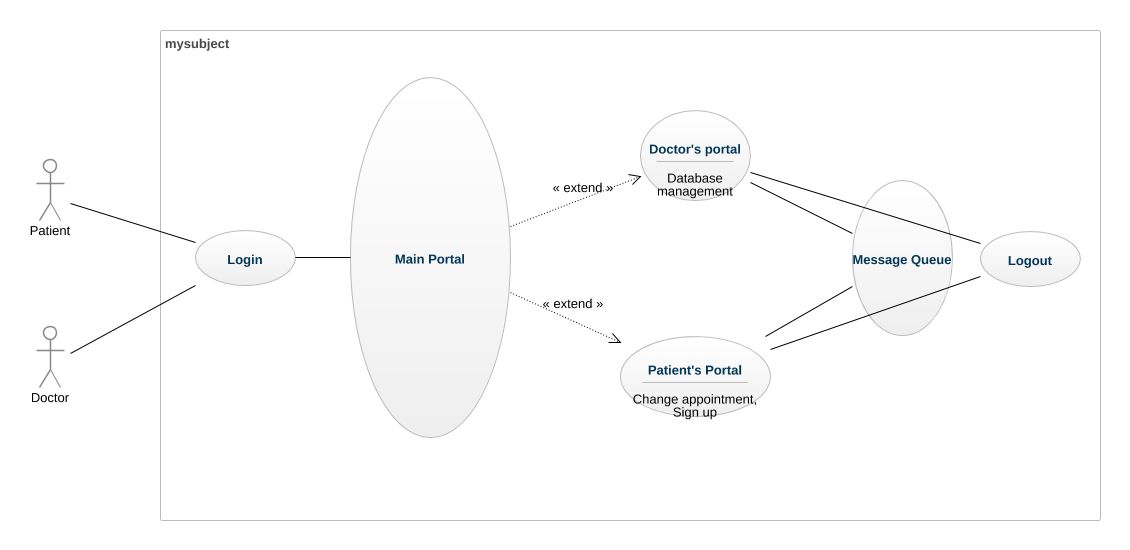
## רשימת ספרות \ ביבליוגרפיה

## תרשימים וטבלאות

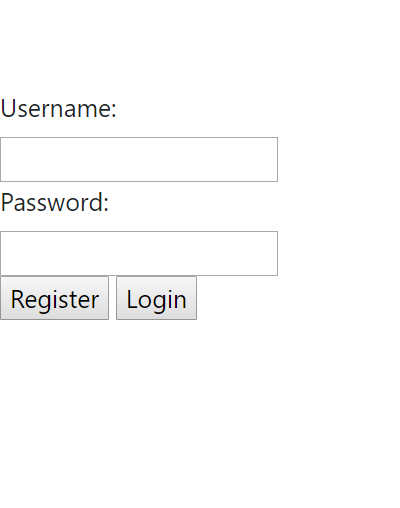
**מסכים (אם לא למעלה)**

**תרשימי תיכון כגון: דיאגרמת רכיבים \ הפצה (UML), דיאגרמת ישויות**

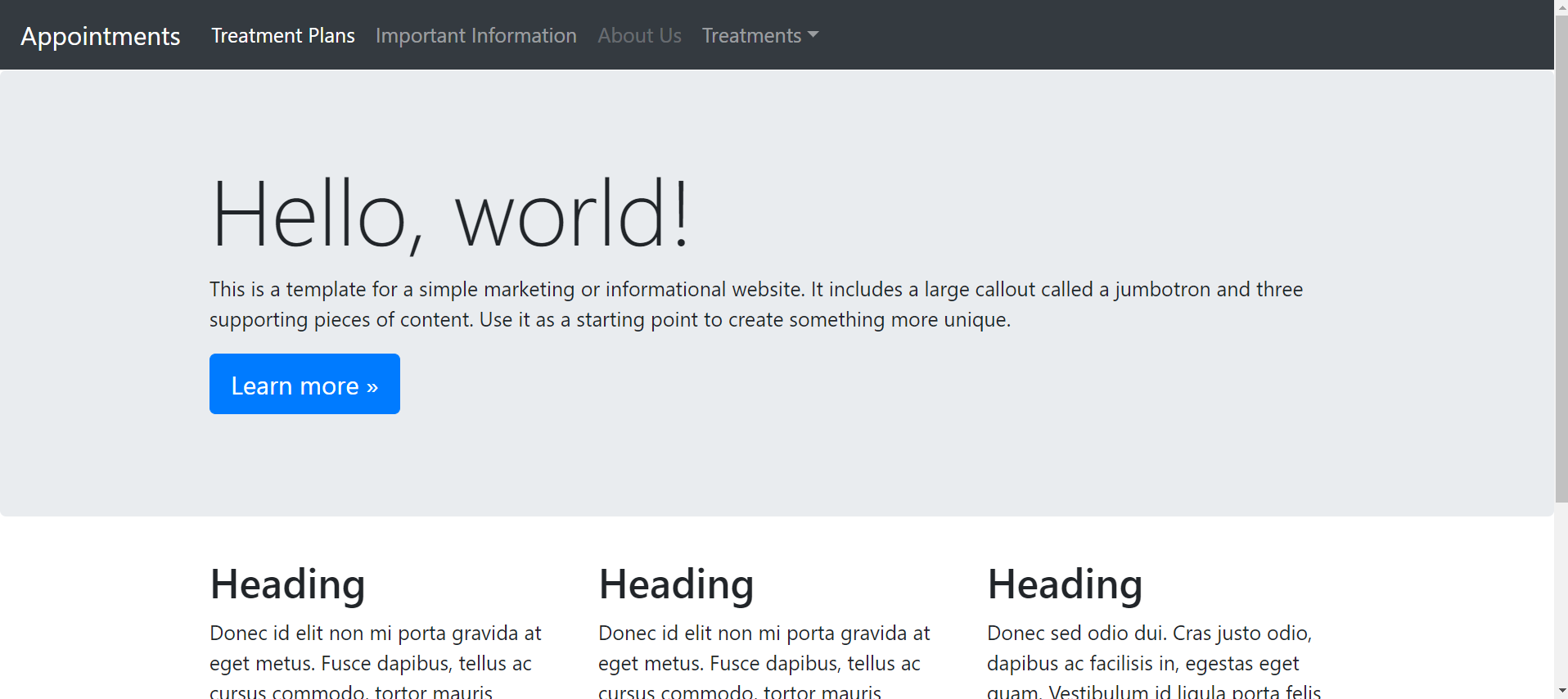
**טבלאות במסד נתונים**

****

מסך ההתחברות:



מסך ראשי:



## תכנון הפרויקט

|  |  |
| --- | --- |
| 15/10 | פגישת היכרות עם הלקוח |
| 22/10 | הגשת טופס התנעה |
| 19/11 | הגשת מסמך הצעה |
| 20/12 | פגישת המשך עם הלקוח |
| 2/1 | הגשת גרסת אלפא |
| 28/2 | פגישה עם לקוח – עיצוב גרפי |
| אמצע סמסטר ב | הגשת גרסת בטא |
| 22/6 | מסירה |
| 24/6 -20/7 | העברה |
|  |  |
|  |  |
|  |  |

## טבלת סיכונים

הסיכונים ממוספרים מ-1 עד 5 כאשר 1 מציין סיכון מזערי ו-5 מציין סיכון ממשי ורציני

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **הסיכון** | **חומרה** | **מענה אפשרי** |
| 1 | שינוי דרישות | 4 | מעקב ותקשורת צמוד עם הלקוח |
| 2 | דרישה טכנית- עבודה עם DB | 5 | אף פעם לא עבדתי עם DB – אתייעץ במידת הצורך עם אנשים שאני מכיר שיכולים לעזור |
| 3 | תקציב | 2 | עלות DB לא גבוהה במיוחד. אני מוכן לכסות |
| 4 | ניהול זמנים | 5 | יש אתגרים בניהול הזמן- אנהל שגרה של עבודה בקצב סביר ואחיד בסגנון SPRINT על מנת להבטיח התקדמות |
| 5 | בעיות DB- אם DB משתנה כך שאין אפשרות לעבוד אתו עוד | 5 | אכין מראש DB נוסף למקרה חירום. אם יהיה צורך אבצע מיגרציה |
| 6 | המערכת צריכה לתמוך בגישה ממכשירים סלולריים | 3 | אומנם יכולים להיות בעיות תאימות תצוגה אבל קיימות פתרונות דוגמת Bootstap שתפקידם לאפשר שימוש סלולרי |
| 7 | מספר דל יחסית של משתמשים לקבלת חוות דעת | 4 | אקבל דיווח כמה שיותר מפורט מהמשתמשים שיש לי |
| 8 | בקשות לקוח בUI | 3 | יש צורך לתאם מול לקוח מראה של האתר כולל שמות לכפתורים |
| 9 | Deployment לשרת רשת | 4 | ייתכן שאצטרך לשנות את הקוד להתאים לשרת חיצוני- במידת הצורך אעזר במומחים כמוכן אשתמש בשרת יציב |

## רשימת\טבלת דרישות

**טבלת דרישות (User Requirement Document)**

|  |  |
| --- | --- |
| מס' דרישה | תיאור |
| 1 | הקמת אתר כללי- LOGIN שונה בהתאם לתפקיד |
| 2 | מתן הרשאות גישה בהתאם לזהות המתחבר |
| 3 | מערכת בקשת ואישור שינוי או ביטול תור |
| 4 | המערכת תשלח את הבקשות שינוי לאישור הרופא |
| 5 | הקמת DB לשמירת נתונים |
| 6 | ניהול DB : הכנסה, מחיקה שאילתות חיפוש |
| 7 | לרופאים תינתן אפשרות לשנות ערכים בDB כגון תוכנית טיפולים ועלות טיפולים |
| 8 | למטופלים יוצג התוכנית טיפול שלהם |
| 9 | מידע שימושי ללקוח- פירוט טיפולים ועלויות |
| 10 | אשפרות גישה מטלפון נייד |
| 11 | שמירת פרטיות הלקוחות על ידי אבטחת מידע |
| 12 | עיצוב גרפי של המערכת שיתאים לצרכי הלקוח |
|  |  |
|  |  |
|  |  |
|  |  |