מפרט דרישות תוכנה - SRS

# מידע כללי

|  |  |
| --- | --- |
| שם הפרויקט: | *ידיד-און YedidOn* |
| צוות הפרויקט | ayeletza, tamarke, hodyama, adinazo, LeahRozenbach |
| מסמכים מצורפים: |  |
| מסמכים קשורים: | הצעת פרויקט, תוכנית פיתוח ,דרישות, מערכת, חוזה, קישורים נוספים |
|  |  |

C:\Documents and Settings\yagel\Desktop\tmp\todo.gifלעשות: מלאו את המידע מעל ומתחת, הוסיפו ומחקו שורות בהתאם לצורך. עדכנו את תוכן העניינים.

תוכן העניינים

[הסטורית שינויים 2](#_Toc275493951)

[1.הקדמה 2](#_Toc275493952)

[1.1מטרה 2](#_Toc275493953)

[1.2היקף 5](#_Toc275493954)

[1.3מילון מונחים 5](#_Toc275493955)

[1.4סקירה 5](#_Toc275493956)

[2.תרחישי שימוש – Use Cases 5](#_Toc275493957)

[3.סיפורי משתמשים – UserStories 7](#_Toc275493958)

[4.דרישות סביבה 7](#_Toc275493959)

[4.1דרישות חומרה 8](#_Toc275493963)

[4.2דרישות תוכנה 8](#_Toc275493964)

[4.3דרישות נוספות 8](#_Toc275493965)

[4.4ממשק משתמש – אב טיפוס 8](#_Toc275493966)

# הסטורית שינויים

|  |  |  |  |
| --- | --- | --- | --- |
| גרסה | תאריך | תיאור השינוי | מקור השינוי |
| 1.0 | ‏20/11/16 | גרסה התחלתית | לא ישים |
|  |  |  |  |

# הקדמה

## מטרה

מוצר התוכנה הינו אתר web הנקרא ידיד- און (YedidOn).

המוצר מפותח עבור וועד ידידי ישיבת "חיי משה" הממוקמת בירושלים וסובלת מקשיי קיום כלכלי.

הוועד שם לעצמו מטרה לעודד תרומות מבוגרי הישיבה תוך שמירה על קשר רציף עימם.

הבעיה: ע"מ לייעל את הליך ההתרמות נדרש לבנות מאגר נתונים לשמירת פרטי הבוגרים, וכן לעקוב אחר פעולות הוועד ותוצאותיהן.

הפתרון: ידיד-און.

המוצר כולל:

1. מאגר נתונים שיכיל את פרטי הבוגרים, את סך התרומות + פירוט תאריכי התרומה ושם התורם. כמו כן, המאגר יכיל פרטי אירועים שונים שנערכו.
2. ממשק נוח למשתמש שיאפשר הרשמה למאגר.
3. תפריטים ופונקציות לניהול שוטף של הוועד.
4. מערכת התראות שונות למנהלים בהתאם לבקשתם, כגון: מספר ימים לפני תום זמן תשלומי תרומה.

המוצר לא כולל:

1. אפשרות לשינוי שפה.
2. אפשרות לתרומה אונליין באתר.

למוצר יהיה קשר לרשימת הבוגרים של הישיבה.

מכיוון שהמוצר נמצא ברשת ה-web והוא לא תוכנה הנמצאת על המחשב, הוא כולל כמה יתרונות :

1. אינו דורש תהליכים מורכבים כלשהם לצורך פריסתם בארגונים גדולים. הדבר היחיד שנדרש הוא דפדפן תואם.
2. המוצר הוא יישום בדפדפן אשר בדרך כלל דורש מעט שטח דיסק בצד הלקוח.
3. המוצר, כמו כל יישומי הרשת, אינו דורש ביצוע של שום פעולת שדרוג בצד הלקוח, מאחר שכל העדכונים מיושמים על השרת ומועברים אוטומטית למשתמשים.

הלקוח: וועד ישיבת "חיי משה".

המשתמשים המיועדים: בוגרי ישיבת "חיי משה", התורמים, הנהלת וועד הישיבה.

## היקף

*גבולות המערכת:*

1. *המוצר ישווק (לבינתיים) רק בשפה העברית.*
2. *המוצר לא יכלול אפשרות לתרומה אונליין.*

תארו את גבולות המערכת, כולל דברים שאינם כלולים במוצר.  
תארו בקצרה את החומרה והתוכנה שהמוצר תלוי בהם או יפעל בעזרתם. במידה ויש פרמטרים משמעותיים של ביצועים או אמינות, תארו אותם גם כן.

## מילון מונחים

חשוב שמתההחלה תהיה הסכמה על מונחים שונים הקשורים למוצר ולפיתוח שלו

הכניסו כאן הגדרות וקיצורי דרך המשמשים בהמשך המסמך

## סקירה

הכניסו כאן סקירה קצרה של תוכן המשך המסמך והתייחסות למסמכים קשורים

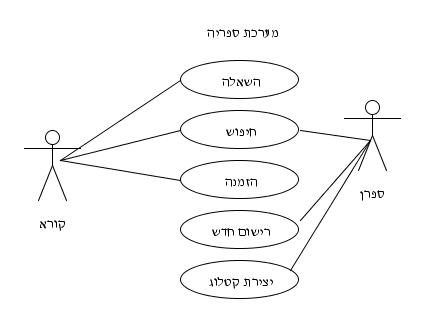
# תרחישי שימוש – Use Cases

תרחישי השימוש מהווים כלי לניתוח הדרישות והבסיס לטבלת הדרישות שעל בסיסה ימשיך פיתוח המוצר.

ספקו לפחות:

1. טבלת שחקנים ובעלי עניין ומטרותיהם

|  |  |
| --- | --- |
| שם שחקן | מטרות |
|  |  |
|  |  |
|  |  |
| בעלי עניין נוספים: |  |
|  |  |
|  |  |

1. תרשים סיכום UML-י של תרחישי השימוש במערכת  
   (אחד לפחות), הכולל את התרחישים העיקריים והשחקנים המשמעותיים במערכת 
2. תרחישי שימוש פורמאליים עיקריים  
   עבור שני תרחישים שנראים לכם החשובים ביותר במערכת (התייעצו גם עם הלקוח, אחד מהתרחישים צריך לכלול את הסיכון העיקרי שזיהיתם ו\או החידוש העיקרי שבמוצר). הם צריכים להיות בסגנון של הדוגמאות שראינו (ראו גם במקורות ההרצאה, למשל Cockburn), במיוחד יש לכלול: שחקן ראשי, תנאי קדם, תנאי סיום מוצלח ונכשל, טריגר, רשימת צעדים של תרחיש ההצלחה, רשימה של הרחבות וטיפול בשגיאות בהתאם. ברור, שאי אפשר לחשוב מראש על כל התקלות האפשריות והפתרונות שלהם, ובכל זאת רשימת ההרחבות צריכה לשקף מחשבה מעמיקה על התרחיש. אם לתקלה מסוימת אין כרגע מענה, יש לציין זאת, ולהסביר מה בכוונתכם לעשות בהמשך.  
   אם יש צורך אפשר לקשר את התרחיש לדיאגראמת פעילות או מסך משתמש.

להלן תבנית לתרחיש עם דוגמא:

|  |  |
| --- | --- |
| שם התרחיש | UC1. הזמנת ספר |
| שחקו ראשי | קורא |
| מטרה | קורא מעוניין לשריין ספר מתוך הקטלוג המקוון |
| היקף ורמה[[1]](#footnote-1) | מערכת הספריה, משתמש |
| בעלי עניין ואינטרסים | קורא – לשריין ספר  בעל הספריה – שרות מורחב לרווחת הלקוחות |
| תיאור | (אם יש צורך, שניים שלשה משפטים המתארים את התרחיש) |
| טריגר | הקורא נכנס למערכת |
| תנאי קדם | הקורא עבר את מסך ההזדהות (login) |
| תנאי סיום מוצלח | הספר שמור עבור הקורא |
| תנאי סיום כישלון | הספר אינו שמור |
| **תרחיש הצלחה עיקרי** | 1. המערכת מציגה מסך ראשי עם קטלוג ואפשרות חיפוש  2. הקורא מזין את שם הספר  3. המערכת מציגה התאמות עם מיקומם  4. הקורא בוחר התאמה ובקשה לשמירה  5. המערכת מאשר את ההזמנה ומציגה את הקטלוג בחזרה |
| הרחבות (שגיאות) | 3א המערכת אינה מוצאת את הספר  3א.1. ... |
| תרחישים חלופיים | 2. הקורא מזין מחבר או נושא |

# סיפורי משתמשים – UserStories

סיפורי המשתמשים מהווים רשימה ברמה כללית של כלל דרישות המשתמשים במערכת הידועות כרגע. מטרתם לתעד את כלל היעדים השונים של הלקוח (כולל דרישות לא-פונקציונליות), באמצעות דוגמאות ברורות כך שניתן יהיה על בסיסם לתכנן ולהעריך את המשך הפיתוח.

כסו את כלל הדרישות הידועות מהמוצר, כולל אלו שפורט בתרחישי השימוש לעיל. מומלץ בשלב זה לתעד את הסיפורים בוויקי במקום במסמך כדי לעודד שיתוף סביבם (במקרה כזה יש להפנות ממסמך זה לדפי הוויקי המתאימים). דרישות פשוטות וברורות מספיק לכסות ע"י שמן (למשל login), אך הציפייה היא גם לדרישות שאותן יש להבהיר באמצעות דוגמאות, כדלקמן.

דוגמא:

|  |  |  |
| --- | --- | --- |
|  | סיפור | תיאור קצר |
| US1 | רכישת דיסק | בתור חובב מוסיקה  אני מעוניין לראות את הכותרים האחרונים  כך שאוכל להזמין ולהנות מהדיסק של... |
| US2 | כניסה לאתר | בתור סטודנט  אני מעוניין להיכנס לאתר הספרייה  כך שאוכל להזמין ספרים מהבית  הערות: יש להפריד בין נושא ה-login לבין פונקצינליות הזמנת ספר  קריטריון קבלה: בהינתן הכתובת של הספריה ושנעשה login מופיע מסך השאלת ספרים |
| US3 | אבטחה | כל המידע הקשור לרכישה יאובטח עם SSL |
|  |  |  |

# דרישות סביבה

תארו את דרישות הסביבה של המוצר. דרישות אלו מתארות את הסביבה הרחבה יותר של המוצר, חומרה, תוכנה, נתונים (בסיסי נתונים) וכדו'. להלן מספר פסקאות לדוגמה:



## דרישות חומרה

## דרישות תוכנה

מוצרי תוכנה, שרותים, חבילות וממשקים (Application Programming Interfaces) שבתכנון לשימוש.

## דרישות נוספות

תקנים נדרשים, **אבטחה**, אמינות, נוחות, עמידות לתקלות  
דרישות לא-פונקציונליות נוספות

## ממשק משתמש – אב טיפוס

 ספקו תרשימים של הרעיון הכללי של ממשק המשתמש. אלו תרשימים של ממשקים עיקריים שיאפשרו למשתמש לבצע את התרחישים העיקריים (ציינו באילו תרחישים מדובר). יש צורך בלפחות תרשים אחד לכל תרחיש (וסיפור משתמש עיקרי) כולל הקשר ביניהם.

אפשר לצייר אותם ביד ולסרוק, ע"י תוכנת מחשב (תוכנה גרפית, PowerPoint) , או לכידת מסך של תוכנת אב-טיפוס שהכנתם. אם ממסך מסוים נפתחת תיבת שיחה או גלילה משמעותית, כדאי לצרף גם אותן. אין צורך בתוצר יפה, אלא משהו משמעותי שמדגים חשיבה על האפשרויות שצריך לקבל המשתמש והמעברים בין המסכים.  
אם אב הטיפוס כבר מספיק מפורט מספיק לצרף צילומי מסך עבורו ובכל מקרה יש לתאר מה מצבו של אב הטיפוס

ממשקי המשתמש משלימים את תרחישי השימוש מכיוון שהם אלו שבעזרתם יפעילו השחקנים את המערכת.

# רשימת דרישות לבירור נוסף

תעדו כאן דרישות ידועות (בד"כ ע"פ מסמך הגדרת הפרויקט) הסותרות דרישות אחרות או דרישות שיש לברר את היקפן

# רשות: טבלת דרישות – User Requirements Definition / Backlog

טבלת הדרישות מקשרת בין צרכי הלקוח לבין מוצר התוכנה המפותח, היא נגזרת מתרחישי השימוש ומקורות נוספים. בשלב זה סיפורי המשתמש יכולים בד"כ להיכנס לטבלה כמו שהם. רשימה זו כוללת בד"כ פירוט ממוספר של כלל הדרישות הפונקציונליות והלא-פונקציונליות של המערכת (לעיתים מפרטים לאחר הטבלה כל דרישה בפסקת טקסט נוספת). כאשר מפרטים תתי משימות, מסמך זה גם נקרא לעיתים Work Breakdown Structure המשמש הלאה להערכה ותכנון.

בפיתוח בשיטת Scrum רשימה זו היא בעצם ה-Backlog של הפרויקט שממנו יילקחו דרישות למימוש עבור כל ספרינט. מכיוון שאנחנו מתכננים לעבור לפיתוח בשיטות אג'ייל, שבהן הניתוח המפורט נעשה תוך כדי המימוש, חלק זה הוא רשות ואם בחרתם לא לעשותו נסתמך על רשימת סיפורי המשתמשים בלבד.

 מלאו טבלת דרישות כדלהלן. לכל דרישה יש לפרט את שמה, סוג הדרישה (פונקציונאלית, אמינות, שמישות, אבטחה, ביצועים, מדרגיות, תחזוקתיות וכדו'), מקור הדרישה (מס' תרחיש, מס' סיפור משתמש או מקור אחר), דרגת הדרישה (1- הכרחית, 2- כדאית או 3- אופציונאלית) והערכת המאמץ למימושה. השאיפה היא לפירוט עד לרמה שיהיה ניתן להעריך את המאמץ לכל דרישה במספר ימי עבודה מועטים שלכם, כך שאם יש דרישה רחבה מדי כדאי לפרוט אותה לתת-דרישות. כדאי לקבץ את הדרישות לקבוצות של דרישות קשורות שצפוי שהעבודה עליהם תהיה כנראה באותו שלב.

הנה דוגמת התחלה של רשימה לפרויקט יומן מקוון, הרשימה שלכם צריכה לכלול כ-20 מאפיינים.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| מס' | דרישה\מאפיין | סוג | מקור | דרגה | מאמץ |
| 1 | צור\מחק יומן חדש | פונקצ' | UC1 | 1 |  |
| 2 | צור\מחק אירוע חדש | פונקצ' | UC2 | 1 |  |
| 3 | צפה ביומן של חבר | פונקצ' | UC2 | 2 |  |
| 4 | הוסף אירוע ליומן חבר | פנקצ' | US1 | 2 |  |
| 5 | יבוא של יומן אאוטלוק | פונקצ' | UC8 | 3 |  |
| 6 | מניעת כניסה למשתמש שאינו חבר | אבטחה | SOW | 1 |  |

תבנית סיכום סקר דרישות תוכנה - SRS Review

 תעדו את מהלך הסקר. את טבלת המשימות נזין בהמשך באתר ניהול הפרויקט כ-issues.

# תוכן

|  |  |
| --- | --- |
| שם הפרויקט הנסקר |  |
| נציגי הלקוח (הסוקרים) |  |
| מועד ומקום הפגישה |  |
| שמות משתתפים |  |

# נקודות שעלו בסקר

|  |  |
| --- | --- |
| 1. | חסר תרחיש חלופי בתרחיש UC2 |
| 2. |  |
|  |  |
|  |  |
|  |  |

# טבלת משימות הנובעות מהסקר (אפשר לחלופין למלא במערכת המשימות github issues)

|  |  |  |
| --- | --- | --- |
| נושא | פעולה נדרשת | באחריות |
| תרחיש UC2 | הוספת תרחיש חלופי במקרה של הזנת פרטים שגויים | אלן טיורינג |
|  |  |  |
|  |  |  |

1. ראו <http://en.wikipedia.org/wiki/Use_case#Design_scopes> [↑](#footnote-ref-1)