שלב 1:

שאלה 4)

על מנת לוודא כי התשובות הנמסרות ללקוח הינן תשובות מהימנות, ראשית נרצה לוודא כי הקלט שהלקוח הקליד נכון ולכן נרצה להעלות בפני הלקוח הודעה שתזכיר לו לבדוק כי הנתונים שהכניס אכן מדויקים.

בין היתר יש להתחשב במקרי הקצה שעלולים להיות, יכול להיות שהלקוח הכניס קלט לא חוקי שנלחץ לו בטעות למשל האות "ר", במקרה כזה נרצה להקפיץ הודעה למשתנה כי הנתונים שעליו להכניס הם רק נתונים מספריים. (ניתן לעשות זאת באמצעות תנאי הבודק האם הקלט שהוכנס הוא מסוג string, במידה וכן נודיע למשתמש במידה ולא נרצה להמשיך בתוכנית החישוב המבוקשת).

מקרה קצה נוסף שיכול להתקיים קשור לבעיה פיזיקלית כפי שידוע זמן הפגיעה בקרקע לא יכול להיות שלילי, ולא שווה לאפס (במידה והטיל משתחרר מגובה 0) ולכן נרצה למנוע מצב בו הזמן שנציב במשוואות היוצרות את הפלט זמן "לא חוקי". על מנת להתגבר על כך נרצה להקים תנאים שיחסמו אפשרות זו.

שאלה 5)

על מנת שהמודל יהיה מדויק יותר נתחשב בגורמים הסביבתיים שהתעלמנו מהם, לדוגמה: ווקטור הרוח (גודל וכיוון), במודל שאנחנו עשינו הקרקע מוגדרת כאחידה (ללא הרים ועוד..) במציאות לא כל הקרקע מוגדרת בגובה 0, חיכוך עם האוויר, מערכת תלת ממדית.

בשביל לשפר את המודל שלנו הייתי מגדילה את הקלט ומתייחסת לעוד נתונים כמו למשל מהירות הרוח (גודל וכיוון) בעת הרצת התוכנית יהיה לנו רלוונטי האם הרוח נגד כיוון מהירות הטיל או עם כיוון מהירות הטיל. אני לא בקיאה מספיק בנוסחאות הפיזיקאיות בשביל להגיד איך וכיצד זה משפיע לכן לא אוכל לכתוב תוכנית מדויקת. אבל כרעיון והגיון אם המהירויות של הטיל והרוח מתלכדת אז המיקום הסופי יגדל ואם מהירויות של הטיל והרוח נגדית את המיקום הסופי יקטן. בנוסף לכך נתחשב בגודל מהירות הרוח ככל שגודל מהירות הרוח יותר גדולה כך תגדל ההסטה של התיל מהמיקום הסופי.

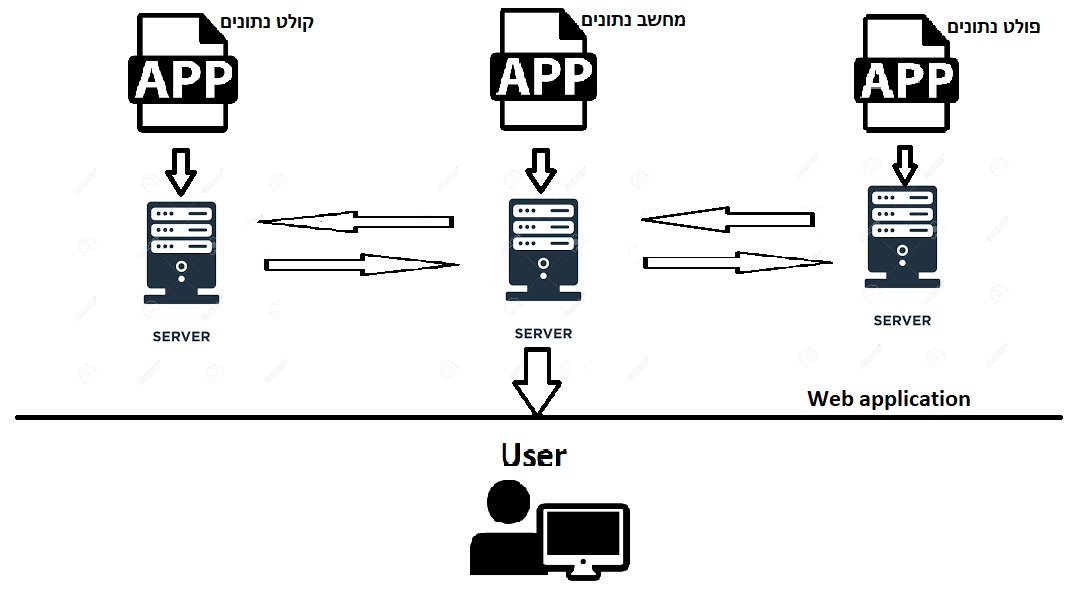
בנוגע לגובה הקרקע שאינו אחיד הייתי מוסיפה לתוכנית מפה אשר מאפשרת לדעת מה גובה הקרקע ביחס לגובה פני הים, ולפי כך לדעת האם זמן הנחיתה של הטיל מתקצר או מתארך לפי גובה הקרקע. כמובן אם בשלב מסוים בעת מסלול הטיל מופיע הר גבוה שיחסום את מסלולו על התכונית להודיע זאת כפלט.

יש לקחת בחשבון כי החיכוך עם האוויר מאט את הטיל, באמצעות נוסחאות פיזיקליות נוכל לדעת כיצד זה משפיע על מהירותו ומיקומו הסופי ונוסיף זאת לתוכנית. נוסף על כך נרצה ליישם את התוכנית במערכת של שלושה צירים (x,z,y).

שלב 2:

שאלה 4)

בהתאם לארכיטקטורת מיקרו סרוויסים נרצה לפתח אפליקציה ראשית המבוססת על שלושה אפליקציות "קטנות" אשר מעבירות ביניהן מידע, הראשונה תחיל את קליטת הנתונים מן המשתמש, השנייה תחיל את החישובים עצמם והשלישית תחיל את האחריות להעניק למשתמש את התוצאות.



שאלה 6)

במידה ונשלחה בקשת חישוב לא צפויה על המערכת להודיע למשתמש כי הנתונים שהקליד אינם תקינים וכי יבדוק את הנתונים שהכניס למערכת ויזין נתונים חדשים.

שאלה 7)

בהתחשב בסיווג המערכת, נוכל להגן על הנתונים שזורמים במהלך התקשורת בעזרת שדרוג בקשת HTTP לבקשת HTTPS.

שאלה 8)

לדעתי ניתן לייעל את המערכת לאור ביקוש חישוב זהה מאת כמה משתמשים, בעזרת מאגר זיכרון מטמון שמאפשר גישה מהירה לפיסת זיכרון (במקרה זה חישובים בעלי נתונים זהים) ומדפיס את התוצאה המתבקשת בצורה מיידית ללא צורך בחישוב. דבר המאפשר הורדת לחץ על השרתים.

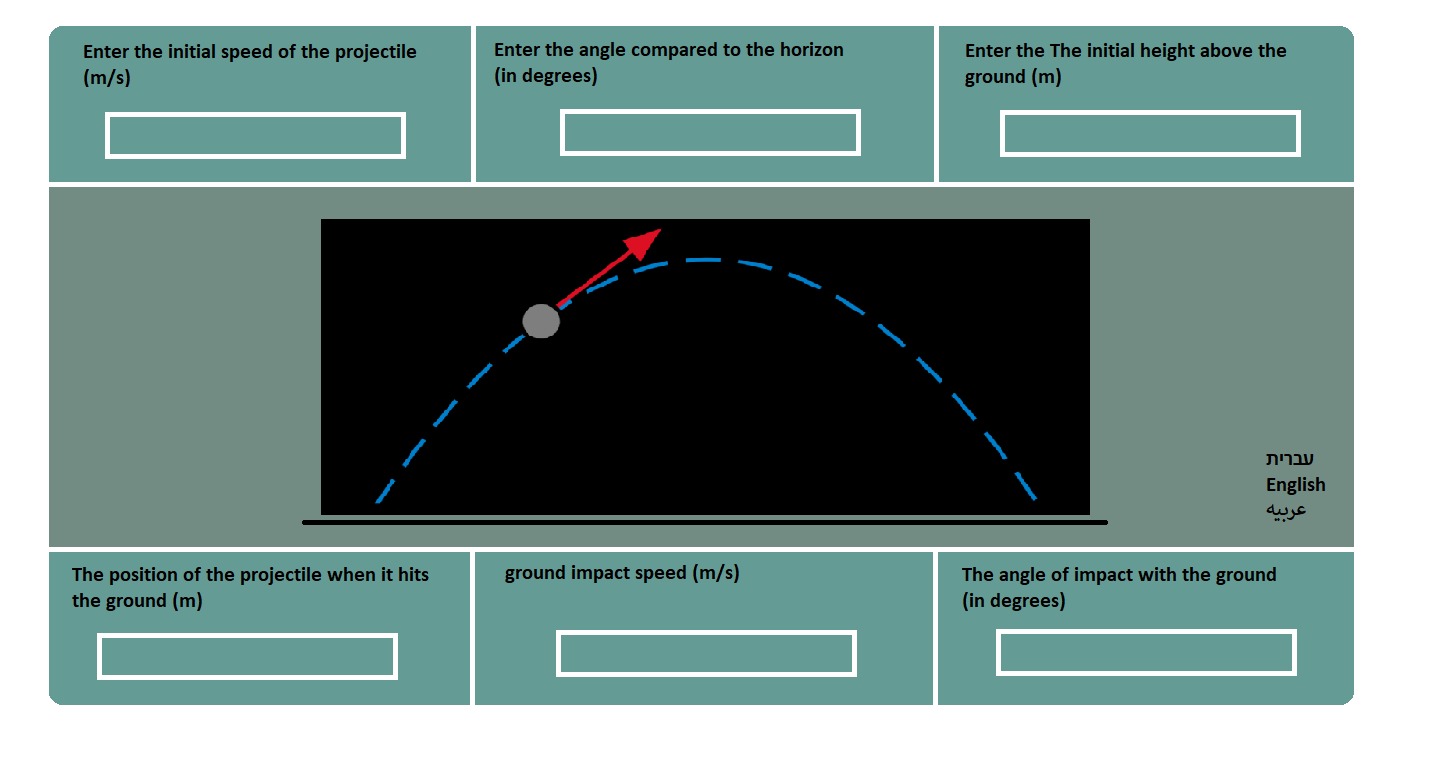
{הדגמה לזיכרון מטמון מתוך אתר בלוג: אם היו שואלים אתכם 'מהו הסכום המכפלה של 5 x 4?', הייתם יודעים שהתשובה היא 20. לא הייתם צריכים לבצע חישוב מתמטי, ביצעתם את החישוב לשאלה זו כל כך הרבה פעמים בחייכם, כך שאתם לא צריכים לבצע זאת שוב – אתם פשוט זוכרים את התשובה לשאלה מבלי להכניס את המוח שלכם למצב "עיבוד נתונים \ חישוב". למעשה, זהו סוג של זיכרון מטמון שאנחנו מפעילים}.

שאלה 9)

נוכל לשפר את המערכת כך שתוכל לתת מענה מהיר למשתמשים רבים בעזרת הגדלת השרתים או לחילופין שדרוג השרתים.

שלב 3:

שאלה 3)



שאלה 4)

בעת פיתוח ממשק המשתמש נשים דגש על פשטות האפליקציה כך שכל אחד יוכל להשתמש בה, בעיניי על אף שהשפה המדוברת במדינה היא עברית יש להנגיש את האפליקציה בשלוש שפות עברית, אנגלית וערבית כך שהמשתמש יוכל לבחור את שפת השימוש באפליקציה. נוסף על כך יש להבהיר למשתמש את יחידות המידה של הערכים שעליו להכניס לאפליקציה מהירות במטר לשנייה, גובה במטרים וזווית במעלות (אולי יש צורך לייעל את הקוד ולהמיר בעצמו את יחידות המידה בהתאם לנוחות המתבקשת לדוגמה: את המרחק הסופי אולי נרצה לציין בקילומטרים, כנ"ל לגביי המהירות שנרצה להכניס בקילומטר לשעה).