



פרויקט Zugspitze Base Alpha

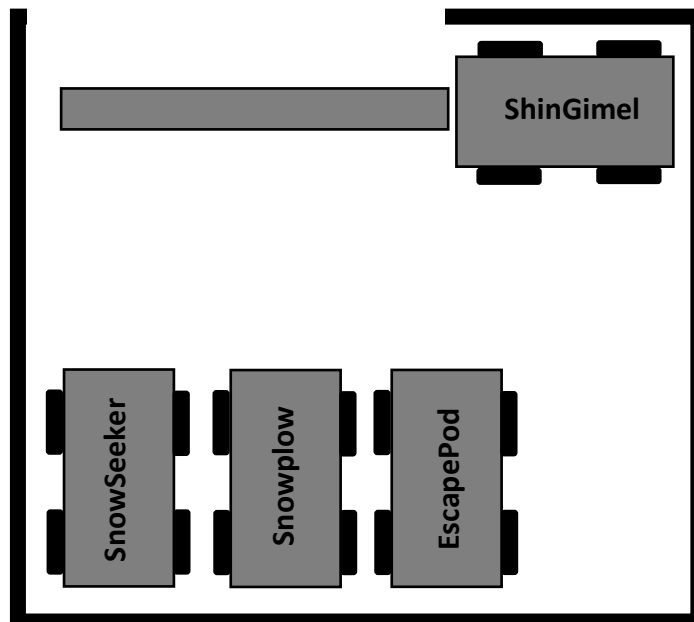
סיפור הרקע:

על הפסגה המושלגת של הר ה Zugspitze הוקם מחנה בו 4 רובוטים אוטונומיים המסוגלים לאסוף מידע על מצב המחנה ולבצע פעולות תחזוקה של המחנה.

מפקד המחנה סובל מקור והחליט לשלוט על מצב המחנה מבקתה באיים המלדיביים. לצורך כך הזמין מערכת שליטה מבוססת IoT שתפעיל באופן אוטומטי את הרובוטים. המערכת צריכה לכלול ממשק משתמש אשר יציג את הנתונים הנמדדים במחנה ואת מצב הרובוטים. כמו כן יאפשר הממשק להפעיל ידנית את כל אחד מהרובוטים.

רשימת הרובוטים:

1. רובוט נייד Snowplow האחראי על פינוי השלג משביל הגישה ועל מדידת הטמפרטורה Temperature. (יש לתכנן ולהדפיס עבורו ב 3D כף לפינוי שלג)
2. רובוט נייד SnowSeeker האחראי על בדיקה פעילה של גובה השלג בכניסה למחנה Snow_Hight. (יש לתכנן ולהדפיס עבורו ב 3D מתאם לחיבור חיישן IR)
3. רובוט נייד ShinGimel האחראי על פתיחה וסגירה של שער המחנה ועל מדידת מהירות הרוח Wind_Speed (יש צורך להדפיס ב 3D כפות מתאימות לחיישן הרוח).
4. רובוט נייד EscapePod האחראי על פינוי חרום של צוות המחנה ועל מדידת מצב הראות (צבע השמיים) Sky_Color. (יש לתכנן ולהדפיס עבורו ב 3D מתאם לחיבור חיישן צבע)



כל אחד מהרובוטים צריך:

1. לשלוח כל שניה את הנתונים שהוא מודד.
2. לשלוח כל שניה דיווח על מצבו.
3. להמתין לפקודת הפעלה ממערכת השליטה ולפעול בהתאם.

מערכת השליטה צריכה לדאוג לתנאים הבאים:

1. כל 5 דקות, לשלח את רובוט מדידת גובה השלג.
2. אם גובה השלג מעל 5 ס"מ יש לשלח את רובוט פינוי השלג.
3. בהילחץ לחצן מצוקה הממוקם על רובוט החילוץ (בכל תנאי מזג אוויר) יש לפתוח את השער לשלח את רובוט פינוי השלג ואחריו את רובוט החילוץ.
4. אם הטמפרטורה נמוכה מהמותר, מהירות הרוח גבוהה מהמותר או מצב הראות גרוע (לבן או שחור) יש להמתין לשובם של כל הרובוטים אשר בפעילות, לסגור את שער המחנה ולהפסיק לשלח רובוטים.

ממשק המשתמש צריך לאפשר

1. להציג את מצב הנתונים הנמדדים.
2. להציג את מצב כל אחד מהרובוטים הניידים (פועל, נח או לא מתקשר).
3. לקבוע ערכים לטמפרטורה ומהירות רוח מותרים.
4. לשלח ידנית את כל אחד מהרובוטים.

משימתכם (אם תחליטו לקבל אותה) בארבעה צוותים:

- לתכן ולהדפיס ב 3D את החלקים הדרושים לרובוטים.
- ליצור את ה Thing וה Mashup הדרושים.
- לתת אפיונים (ספציפיקציות) לתכנות הרובוטים.
- לבצע אינטגרציה של כל המרכיבים ולבדוק תקינות המערכת.

הנחיות עבודה:

הקמנו עבורכם Thing בשם `216015_Zug_Thing`

כל צוות יקים בתוך Thing זה את ה Properties הבאים הרלוונטים לצוות:

1. `X_Status` (X הוא שם הרובוט למשל `ShinGimel_Status`) יקבל את מספר שלם המציין את מצב הרובוט (0 בהמתנה למשימה, 1 במשימה)
 2. `X_Command` (למשל `SnowSeeker_Command`) יקבל את מספר שלם המציין את סוג המשימה שהרובוט צריך לבצע (0 להמתין 1 לצאת למשימה 1, 2 לצאת למשימה 2)
 3. `Y` שם הפרמטר הנמדד ע"י הרובוט למשל `Temperature` יקבל מספר שלם המציין את הערך הנמדד.
- כל צוות יקים בתוך Thing זה את ה Services הרלוונטיים כדי לשרת את המשימה של הרובוט שלהם, כלומר אשר הפעלתם תבצע קריאה של ערכים או השמה שלהם מ ואל ה Properties השייכים להם.

הקמנו עבורכם Mashup בשם `216015_Zug_Mashup`

כל צוות ימקם Widgets הרלוונטיים כדי לשרת את המשימה של הרובוט שלהם ויקשר אותם לפי הצורך.

אין צורך לממש את פונקציות מערכת השליטה כי אם רק להכין את ה Services שהיא תשתמש בו עבור הרובוט שלכם.

בהצלחה!

