

עליכם להוסיף לפרויקט שלכם שכבה של **Business Logic**, שבשלב זה תמומש בצורה בסיסית מעל שכבת **DAL - DATA ACCESS LAYER**.

יש לעדכן את מחלקות ה-**RestController** שכתבתם ב-sprint הקודם, כך שהן תפננה למחלקות החדשות שעליכם להוסיף באמצעות ממשקים (java interfaces).

בשלב זה, שכבת הלוגיקה מאפשרת פעולות בסיסיות לניהול נתונים - שמירה, שליפות פשוטות, עדכון ומחיקה.

שימו לב לדגשים הבאים לגבי המימוש שלכם:

- עליכם להגדיר את שם ה-**domain** שלכם, באמצעות משתנה בשם **spring.application.name** שיוגדר בקובץ **application.properties**
 - שימו לב כי אין להגדיר את שם הפרויקט, או שם ה-**domain**, באמצעות קבועים בקוד עצמו
- החליפו את תלויות הפרויקט שלכם, בקבצי ה-JAR שזמינים לכם תחת **קבצי תלויות לפרויקטי IoC עם תמיכה בממשק WEB ובסיסי נתונים רלציוניים**
 - עליכם להוסיף לקובץ ה-**application.properties** הגדרות להדפסת שאלות ה-SQL ל-log של הפרויקט שלכם
- תיאור המפרט של ממשקי שכבת ה-business logic מופיע בקישור הבא **Sprint 3 Business Logic Spec**
 - יש להקפיד שהמימוש שלכם יתאים למפרט, כפי שהוא מופיע בתרשים, כולל שמות מדויקים ל-packages, interfaces, enums, methods.
 - למשל, יש להקפיד שה-enum שתוסיפו יקרא **UserRole** ויופיע ב-package ששמו **iob.data**
 - למשל, יש להקפיד שהממשק **UserService** יופיע ב-package ששמו **iob.logic** וחתימת המתודות שלו תהיה בדיוק כפי שמתואר במפרט
 - פרמטרים לממשקים, כמו **UserBoundary** הם מסוג המחלקות שהגדרתם כבר ב-sprint הקודם
 - המחלקות החדשות המממשות את הממשקים, אמורות לאחסן, בשלב זה של הפרויקט, את העצמים שהן מנהלות ב**בסיס נתונים רלציוני H2 EMBEDDED**
 - שימו לב כי את אחסון הנתונים יש לבצע באמצעות עצמי **Entity**
 - אתם יכולים לעדכן את המאפיינים למחלקות ה-**Entity** כראות עינכם, ובלבד שיאפשרו לאחסן בבסיס הנתונים את כל הנתונים שנתמכים בממשק ה-REST
 - יש להקפיד להשתמש בטרנזקציות כשפונים לבסיס הנתונים
 - שימו לב, כי בשלב זה, המחלקות שמממשות את ה-Services אינן משפיעות זו על זו

- כך למשל, המחלקה שתממש את InstancesService אינה צריכה, בשלב זה, לבדוק את ה role של ה user-שמפעיל את המתודות שלה, או לבדוק אם קיים בכלל user כזה במערכת
- בשלב זה של הפרויקט, המימוש של ה Services-יבצע אימות מינימאלי של הקלט שלו -
 - למשל, יש לוודא שהקלטים והמאפיינים החשובים שלהם אינם null,
 - למשל, יש לוודא כי role של משתמשים מכיל ערכים תקינים,
 - יש לוודא שכתובות הדואל של משתמשים חדשים הן אכן כתובות תקינות,
 - יש לוודא כי שמות המשתמשים וה Instances-אינם null או מחרוזת ריקה,
 - יש לוודא כי ה avatar-של משתמשים איננו null או מחרוזת ריקה,
 - יש לוודא כי ערכי ה Instance של Activity אינם null או מחרוזת ריקה
 - יש לוודא כי אובייקטים של Activities כוללים בהכרח פרטים של משמשים שקיימים בבסיס הנתונים שהפעילו אותם,
 - בנוסף, יש לוודא כי אובייקטים של Activities כוללים בהכרח מזהה של Instance שהם מופעלים עליו, שקיים בבסיס הנתונים
- כשה Services-שומרים עצמים חדשים במערכת, עליהם לעדכן שדות שבשליטת השרת:
 - יש לעדכן את ה domain של עצמים חדשים, כך שיהיה זהה לשם הפרויקט שלכם
 - יש לעדכן תאריכי יצירה
 - יש לייצר ערכי id מתאימים ל INSTANCES ול ACTIVITIES-חדשים, כך שיאפשרו הפעלות בו זמנית של פעולות יצירת עצמים חדשים ורישומם בבסיס הנתונים
- כאשר פונים ל UsersService-כדי לעדכן פרטי משתמשים במערכת באמצעות המתודה updateUser שימו לב כי אינכם אמורים לאפשר לעדכן את כתובות הדואל ואת ה domain של משתמשים קיימים במערכת
- כאשר פונים ל InstancesService-כדי לעדכן פרטי Instance במערכת באמצעות המתודה updateInstance שימו לב כי אינכם אמורים לאפשר לעדכן את הפרטים הבאים:
 - לא ניתן לשנות את פרטי ה instanceId-כולל ה domain-שמוגדר לאותו Instance
 - לא ניתן לשנות את פרטי היוצר ומועד היצירה של ה Instance-ששמרתם במערכת כשיצרתם את ה Instance-
- כאשר המימוש של ה Service-מתבקש להחזיר נתונים שאינם קיימים במערכת, עליו להעיד RuntimeException למשל במקרים הבאים:
 - כשמנסים לעשות login למשתמש שאיננו קיים
 - כשמנסים לשלוף instance ספציפי שאיננו קיים במערכת
- בשלב זה של הפרויקט, המימוש של המתודה של ActivitiesService ששמה invokeActivity יכול להחזיר כל עצם שאתם רוצים ושאפשר להפוך אותו ל json-ובנוסף, מתודה זו תשמור

את העצם שקיבלה כפרמטר, לאחר שתיצור לו id ותעדכן את פרטי המשתמש שיצר אותו
ומועד היצירה של activity-

• Sprint 3 Business Logic Spec

Integrative Software Engineering - Specifications for Business Logic - sprint 3

- Implement the following classes, interfaces and enum in your project
- Note the packages defined for each of them
- Make sure you specify correctly the interface signatures
- Make sure you implement the Service Interfaces

