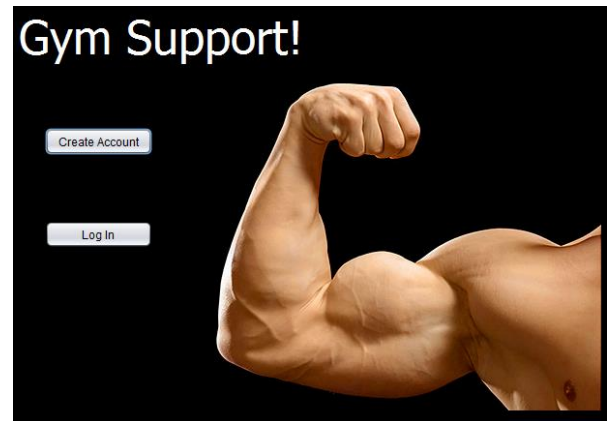# Advanced Software Engineering

I. Participants Names

Τσόγκας Βασίλειος mai19079
Κουφάκης Νικόλαος mai19033
Κατσαρός Μάριος mai19026
Σπανίδης Θεόφιλος mai19068
Μπουγάτσιας Μάρκος mai19050

II. Short Description of our project

Δημιουργία μίας γραφικής επιφάνειας GUI με στόχο την προσέλκυση νέων χρηστών και μελών για εγγραφή σε ένα εικονικό «Γυμναστήριο».Κάθε νέα έκδοση της εφαρμογής μας,δίνει νέες δυνατότητες και προοπτικές. Από την απλή καταχώριση δεδομένων και εγγραφή στη βάση μας,στην αναζήτηση νέας προπόνησης και στην εμφάνιση πληροφοριών για το έμψυχο δυναμικό των γυμναστών με την ειδικότητα τους και ένα θερμό μήνυμα καλωσορίσματος.Τέλος μετά την απόκτηση πλήρης πρόσβασης στην εφαρμογή με εγχρηματή εγγραφή,ο χρήστης μπορεί να μαθαίνει το πρόγραμμα των ημερήσιων προπονήσεων, ενημέρωση του προσωπικού status μας,περισσοτέρες φυσικά συμβουλές προπόνησης και διατροφής καθώς και το πιο σημαντικό, το online booking από το κινητό του.

III. URL link of GitHub

a)   https://github.com/mai19079/GymSupport

b)   https://github.com/mai19079/GymSupport/blob/master/RunGymSupport.zip

c) **Installation Instructions**

    1.1.1.   Unzip το RunGymSupport.zip

    1.1.2.   Run GymSupport.jar file

d) **Further Documentation**

Required PostgreSQL Database. Instructions made for version 9.4. For a different version, you might need to customize some steps).

DB Initialization :

2. Download gym_support.sql in a temp folder (e.g. C:\temp)
3. Go to your ..\PostgreSQL\9.4\bin loaction
4. Open command window here (Shift _ RClick)
5. run: createdb -h localhost -p 5432 -U postgres -E utf8 gym_support
6. run: psql -h localhost -p 5432 -U postgres -d gym_support -f c:\temp\gym_support.sql
7. OPTIONAL: connect to localhost via pgAdmin and administer gym_support DB

IV. **Functional Requirements**

a) **User Stories**

USER STORY 1.0

As a user I want to be able to register my profile data and find a proper workout.

ACCEPTANCE CRITERIA

A user should:

-see a contact information form where he can insert his first and last name, age,weight,sex and etc.

-be able to select from a limited set of workouts and trainings sessions.

EDGE CASES

-If there is an invalid data entry, the user should see an error message.

-If a mandatory field isn't filled, the application should display a message and won't allow the user to continue until he completes the filling in.

USER STORY 1.1

As a user I want to be able to select either if i want to Login or to create a New Account

ACCEPTANCE CRITERIA

A user should be able to:

-see a contact information form where he can insert his/her data and then select submit in order to finish his/her free subscription.

-login with his/her credentials (username, password) if he is already subscribed.

EDGE CASES

-If there is an invalid data entry, the user should see an error message.

-If a mandatory field isn't filled, the application should display a message and do not allow the user to complete the filling in.

USER STORY 2.0

As a user i want to be able to select among exercise programs and fitness tips.

So that i can find the ones that suits me or i like the most.

ACCEPTANCE CRITERIA

A user should be able to:

- learn more details about a training class.

- get limited tips and exercise propositions according to his/her current athleticism and status (lean/bulk).

EDGE CASES

-if he select a program with different goal, the user should see a message: "You have selected a bulking training session but your current status is lean"

USER STORY 2.1

As a user i want to learn more before i select among exercise programs and fitness tips.

ACCEPTANCE CRITERIA

A user should be able to:

-learn more details about a training class.

-get more tips and workout proposals from gymnasts with specialization in bulking or leaning.

USER STORY 2.2

As a user i want to be able to start a full subscription.

So that i can get full access and explore all my benefits and provisions of this gym application.

ACCEPTANCE CRITERIA

A user should be able to:

- check his/her subscription fee, duration and expiration date.

-find out more about his/her trainer expertise.

-update his/her goal and training status (Lean/Bulk)

-get unlimited training and fitness tips.

EDGE CASES

-if the subscription is or about to expire, get a notification message of renewing.

USER STORY 2.3

As a user i want to be able to start a full subscription.

So that i can get full access and explore all my benefits and provisions of this gym application.

ACCEPTANCE CRITERIA

A user should be able to:

- check his/her subscription fee, duration and expiration date.

-find out more about his/her trainer expertise.

-update his/her goal and training status (Lean/Bulk)

-get unlimited training and fitness tips.

EDGE CASES

-if the subscription is or about to expire, get a notification message of renewing.

USER STORY 3.0

As a user i would like to get informed about the availability of training sessions in the fitness center.

ACCEPTANCE CRITERIA

A user should be able to:

-learn the timetable of all the training sessions available.

USER STORY 3.1

As a user I want to be able to update my profile, so that my suggested workout can evolve.

ACCEPTANCE CRITERIA

A user should be:

-able to update his/her profile and physical status (weight, injury).

-able to select either he/she wants to get bulked or leaned.

-be able to get more tips and training session proposal according to his/her selection.

EDGE CASES

-If he/doesn't select bulk or lean the user should be able to see all the suggested exercises and tips.

USER STORY 3.2

As a user I want to be able to book a training session or a workout and read my coaches tips on their profiles.

ACCEPTANCE CRITERIA

A user should be able:

-book a spot for a training class.

-Read his/her coaches training tips.

EDGE CASE

-if there is not an available spot, the user should see a message of fullness or unavailability.

-if he select a program with different goal, the user should see a message: "You have selected a bulking training session but your current status is lean".

-if a coach hasn't upload any tips, the user should see a message: "You can contact me in person during my working hours."

Task Name	User Story	Product Backlog Item	Story	Sprint Ready	Priority	Status	Story Points	Assigned to Sprint
Sprint 1						**Complete**	**24**	
Task 0	As a gym addict, i want to have an application with graphical user interface that provides me with workout instructions and training tips.	Prepare main window	Yes	Yes	High	Complete	8	Yes
Task 1.0	As a user I want to be able to register my profile data and find a proper workout.	Enhance UI with Pprofile Data Window. Add Simple Workout Plans.	Yes	Yes	Medium	Complete	8	Yes
Task 1.1	As a user I want to be able to select either if i want to Login or to create a New Account	Add LogIn Dialog, Move CreateProfile to Dialog. User data saved in JSON Files.	Yes	Yes	Medium	Complete	8	Yes
Sprint 2						**Complete**	**40**	
Task 2.0	As a user i want to be able to select among exercise programs and fitness tips. So that I can find the ones that suits me or I like the most.	Adding RDBMS Support (PostgreSQL). Prepare data and Insert into DB.	Yes	Yes	High	Complete	10	Yes
Task 2.1	As a user i want to learn more before I select among exercise programs and fitness tips.	Enhance UI with Ineractive Workout Window.	Yes	Yes	High	Complete	10	Yes
Task 2.2	As a user I want to be able to start a full subscription. So that i can get full access and explore all my benefits and provisions of this gym application.	Add Subscription functionality wit registration code. (Payment and code acquisition via Bank Account/SMS/Email)	Yes	Yes	High	Complete	10	Yes
Task 2.3	As a user i want to be able to start a full subscription. So that i can get full access and explore all my benefits and provisions of this gym application.	Add Trainers, Tips and Various workout plans.	Yes	Yes	Medium	Complete	10	Yes
Sprint 3		**Code Freeze - Refactor**				**In Progress**	**0**	
Task 3.0	As a user i would like to get informed about the availability of training sessions in the fitness center.	Add Gym Facilities Schedule. Update status via web services.	Yes	Yes	Low	Not Started	10	Yes
Task 3.1	As a user I want to be able to update my profile, so that my suggested workout can evolve.	Record Somatometric Data, Calculate BMI, Create/Implement algorithms adjust Workout Plan	Yes	Yes	Low	Not Started	20	Yes
Task 3.2	As a user I want to be able to book a training session or a workout and read my coaches tips on their profiles.	Connect to Gym's booking app.	Yes	Yes	Low	Not Started	10	Yes

UserList

-instance: UserList
-userList: new ArrayList<User>()
--
+getUserList() :ArrayList<User>
-UserList()
+addUser(u : User) : void
+find(username : String, password : String) :
gym.User
+checkLoginUser(username : String) : int
+checkUser(username : String) : User

PGClass

-instance : PGclass
-url : String = "jdbc:postgresql://localhost/gym_support"
-user : String = "postgres"
-password : String = "sys"
-conn : Connection
--
-PGClass()
+connect() : Connection
+getConnection(0 : Connection
+executeSelectQuery(q : String) : ResultSet
+executeUpdateQuery(q : String) : int

WorkoutPlan

-currentUser : User
-programid : String
-name : String
-caloriesperday : String
-daysoftraining : String
-trainer : String
-workout : Map
--
+WorkoutPlan(programid : String, name : String, caloriesperday : String, daysoftraining : String, trainer : String, workout : Map)
+getCalories() : String
+displayWorkout() : String
+info() : String

User

-username : String
-mail : String
-name : String
-surname : String
-age : int
-gender : String
-height : int
-weight : int
-pass : String
-fullSubscription : int
--
+User(username : String,..)
+User(username : String,...fullSubscription : int)

MiddlewarePostgreSQL

-instance : MiddlewarePostgreSQL
-expertise : Map<String, String> = new HashMap<>()
-trainers : Map<String, Trainer> = new HashMap<>()
-workout : Map<String, String> = new Hash<Map<>()
-exercise : Map<String, String> = new HashMap<>()
--
-MiddlewarePostgreSQL()
+saveUser(username : String,...) : User
+checkLogin(user : String) : User
+setupLists() : void
+createUser() : User

FreeWorkout

-workoutList : ArrayList<WorkoutPlan> = new ArrayList()
--
+FreeWorkout(mainFrame : GymSupportUI, modal : boolean)
-loadWortiout(workoutFile : String) : void
-initComponents() : void
-closeBtnActionPerformed(evt : ActionEvent) : void
-leanChoiceActionPerformed(evt : ActionEvent) : void
-bulkChoiceActionPerformed(evt : ActionEvent) : void

Workout

~mainFrame : GymSupportUI
~currentUser : User
--
+Workout(mainFrame : GymSupportUI, modal : boolean)
-initComponents() : void

IMiddleware
<<Interface>>

+checkLogin(user : String) : User
+saveUser(username : String,...) :User
+createUser() : User

MiddlewareJSON

-instance : MiddlewareJSON
--
-MiddlewareJSON()
+checkLogin(user : String) : User
+saveUser(username : String,...) : User
+createUser() : User

FullWorkout

~expertiseListModel : DefaultListModel = new DefaultListModel()
~workoutListModel : DefaultListModel = new DefaultListModel()
~exerciseListModel : DefaultListModel = new DefaultListModel()
~trainerComboModel : DefaultComboBoxModel<String> = new DefaultComboBoxModel<String>()
-expertiseList : JList<String>
-workoutPlanList : JList<String>
--
+FullWorkout(mainFrame : GymSupportUI, modal : boolean)
-getTrainersByExpertise(expertise : String) : List<Trainer>
-getWorkoutByExpertise(expertise : String) : List<String>
-initComponents() : void
-expertiseListValueChanged(evt : ListSelectionEvent) : void
-trainerComboActionPerformed(evt : ActionEvent) : void
-workoutPlanListValueChanged(evt : ListSelectionEvent) : void
-getExerciseByName(wplan : String) : List<Exercise>

GymSupportUI

-currentUser : User = null
--
+GymSupportUI()
+getProfilePanel(0 : JPanel
+writeProfileInfo(u : User) : void
+setFullSubscriptionStatus(b : boolean) : void
-initComponents() : void
-createAccountBtnActionPerformed(evt : ActionEvent) : void
-logInBtnActionPerformed(evt : ActionEvent) : void
-updateProfileBtnActionPerformed(evt : ActionEvent) : void
-requestWorkoutActionPerformed(evt : ActionEvent) : void
-genderComboActionPerformed(evt : ActionEvent) : void
-weightTextActionPerformed(evt : ActionEvent) : void
-fullSubscriptionBtnActionPerformed(evt : ActionEvent) : void

Trainer

-code : String
-name : String
-gender : String
-profile : String
--
+Trainer(code : String, name : String, gender : String, profile : String)

LoginScreen

-mainFrame : GymSupportUI
--
+LoginScreen()
+LoginScreen(mainFrame : GymSupportUI)
-initComponents() : void
-logInBtnActionPerformed(evt : ActionEvent) : void

Exercise

-id : int
-code : String
-name : String
-descr : String
-comments : String
--
+Exercise(id : int, code : String, descr : String, comments : String)
+Exercise(name : String)

CreateAccountUI

-mainFrame : GymSupportUI
-instance : CreateAccountUI
--
-CreateAccountUI(mainFrame : GymSupportUI)
+getInstance(mainFrame : GymSupportUI) : CreateAccountUI
-initComponents() : void
-registerBtnActionPerformed(evt : ActionEvent) : void
-genderComboActionPerformed(evt : ActionEvent) : void
-weightTextActionPerformed(evt : ActionEvent) : void

creates
references