FiTracks
Software Architectural Design

Submitted to:

Asst. Prof. Ma. Rowena C. Solamo
Faculty Member
Department of Computer Science
College of Engineering
University of the Philippines, Diliman

Submitted by:
Aguilana, Trina B.
Manguiat, Glenn Karlo D.
Villanueva, Ian N.

In partial fulfillment of Academic Requirements
for the course
CS 191 Software Engineering I
of the
1st Semester, AY 2018-2019

Unique Reference:

The documents are stored in the https://fitracks.wordpress.com in the Project Deliverables Section or can be accessed through http://bit.ly/FiTracksDesignEngineering .

Document Purpose:

This document is provided to show the program specification of a particular controller class of our health and fitness record management system.

Target Audience:

This document also serves its purpose for Asst. Prof. Rowena Solamo, her students, and others analyzing the data design of the application and its components.

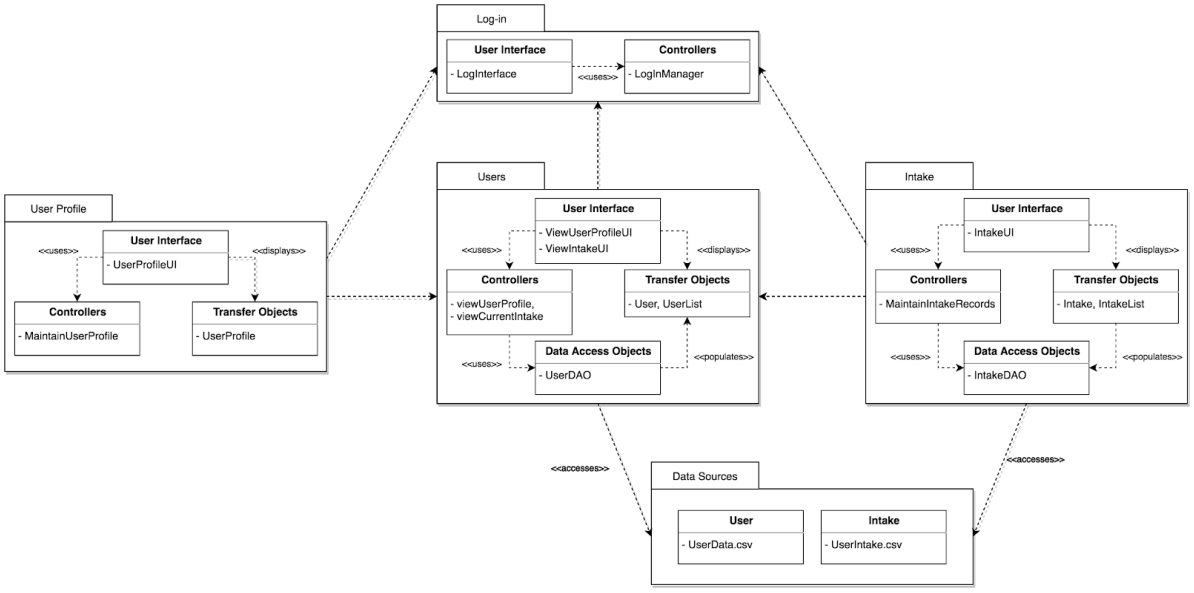
Revision Control:

Revision Date	*Person Responsible*	*Version Number*	*Contribution/Modification*
10/29/18	Ian N. Villanueva	1.0	Initial Document; Added Unique Reference, Document Purpose, Target Audience, System Name, Description.
11/03/18	Glenn Karlo D. Manguiat	1.1	Revised Software Architecture Model; Data Sources Package, DAO Packages;
11/03/18	Ian N. Villanueva	1.2	Added Screen Names and Controllers Package (Controller Names and Description).
11/05/18	Ian N. Villanueva	1.3	Added names on DAO packages and Transfer Object Packages descriptions.
11/05/18	Trina B. Aguilana	1.4	Added User Interface Package and Controllers Package Responsibilities

***System Name*:** FiTracks - Health and Fitness Record Management System

Description: In our health and fitness record management system, a user maintains a personal food and water intake tracker upon log-in. User establishes a profile which contains his/her personal details, current height, current weight, and current health condition, if with complications or none. The establish profile may be viewed and updated at any time possible. User can input meal intake and water intake, which may be updated and deleted upon addition. Lastly, the current meal and water intake status can be viewed by the user in different time schemes.

***Revised Software Architecture Model*:**



User Interface Package:

Screen Name	Description
LogInInterface	This is the screen where the user logs in to the system *Attributes:* String username String password *Responsibilities:* public void InputCredentials(String username, String password) public void LogIn()
IntakeUI	This is the screen that modifies an intake to the database *Attributes:* String WaterInake String watercontainer Float amount Date date_consumed String time_consumed String FoodIntake String foodname int foodintakeid int waterintakeid *Responsibilities:* public void ChooseIntakeData(String WaterIntake) public void EnterIntakeData(String watercontainer, float amount, Date date_consumed, String time_consumed) public void SubmitIntakeData(String watercontainer, float amount, Date date_consumed, Date time_consumed) public void ChooseIntakeData(String FoodIntake) public void EnterIntakeData(String foodname, float amount, Date date_consumed, String time_consumed) public void SubmitIntakeData(String foodname, float amount, Date date_consumed, Date time_consumed) public void ChooseIntaketoEdit(int foodintakeid) public void ChooseIntaketoEdit(int waterintakeid)

	public void ChooseIntaketoDelete(int foodintakeid) public void ChooseIntaketoDelete(int waterintakeid)
UserProfileUI	This is the interface of user whenever they want to view or edit their respective user profiles *Attributes:* private int userID private float weightInKilograms private float heightInCentimeters private String ContactNumber private String Email private String Username private Date Birthdate private int healthConditionID private String healthCondition *Responsibilities:* public void ChooseWeightOrHeight(String edit) public void editWeight(float weightInKilograms) public void editHeight(float heightInCentimeters public void saveEditHeight(float heightInCentimeters) public void saveEditWeight(float weightInKilograms) public void AddHeightWeight(float heightInCm, float weightInKg) public void ChoosePersonalDetailtoEdit(int PersonalDetailID) public void AddPersonalDetails(int PersonalDetailID) public void EnterPersonalDetailsData(Date Birthdate, String ContactNumber, String Email,String Username) public void SubmitPersonalDetailsData(Date Birthdate, String ContactNumber, String Email,String Username) public void AddHealthCondition(String healthcondition) public void AddHealthCondition(int healthconditionID) public void ChooseHealthConditiontoEdit(int healthconditionID) public void ChooseHealthConditiontoDelete(int healthconditionID) public void saveHealthCondition(String healthcondition) public void view Height(float heightInCentimeters) public void viewWeight(float weightInKilograms)

	public void view HealthConditions(String healthcondition) public void view PersonalDetails(Date Birthdate, String ContactNumber, String Email, String Username)
ViewUserProfileUI	This is the interface for when the user wants to view his or her profile. *Attributes:* int userID *Responsibilities:* public void getUserProfile(int userID)
ViewIntakeUI	This is the interface for when the user wants to view a food or water intake. *Attributes:* *Responsibilities:* public void ChooseIntakeData( )

Controllers Package:

Controller Name	**Description**
MaintainIntakeRecords(abstract)	This is the control that maintains intake records. It is considered an abstract class. *Attributes*: private Intake userIntake; private IntakeList userIntakeList;
AddIntake	This is the control that adds an intake to the system. It extends MaintainIntakeRecords. *Responsibilities*: public bool isIntakeDataComplete(String watercontainer, float amount, Date date_consumed, Date time_consumed) public bool NewIntake(Date date_consumed, Date time_consumed) public void saveIntakeData(String watercontainer, float amount, Date date_consumed, Date time_consumed) public bool isIntakeDataComplete(String foodname, float amount, Date date_consumed, Date time_consumed) public bool NewIntake(Date date_consumed, Date time_consumed) public void saveIntakeData(String foodnamer, float amount, Date date_consumed, Date time_consumed)
EditIntake	This is the control that edits an intake to the system. It extends MaintainIntakeRecords. *Responsibilities*: public void editIntake(int waterintakeid)

	public void editIntake(int foodnameid)
DeleteIntake	This is the control that deletes an intake to the system. It extends MaintainIntakeRecords. *Responsibilities*: public void deleteIntake(int waterintakeid) public void deleteIntake(int foodnameid)
MaintainUserProfile(abstract)	This is the control that maintains user profiles. It is considered an abstract class. *Attributes*: private UserProfile profile;
EditWeightHeight	This is the control that edits user's weight and height to the system. It extends MaintainIntakeRecords. *Responsibilities*: public void editWeight(float weightInKilograms) public void editHeight(float heightInCentimeters) public void AddHeightWeight(float heightInCm, float weightInKg) public void saveEditWeight(float weightInKilograms) public void saveEditHeight(float heightInCentimeters)
EditPersonalDetails	This is the control that edits user's personal details to the system. It extends MaintainIntakeRecords. *Responsibilities*: public bool checkData(Date Birthdate, String ContactNumber, String Email,String Username) public void EditPersonalDetailsData(Date Birthdate, String ContactNumber, String Email, String Username) public void SavePersonalDetailsData(Date Birthdate, String ContactNumber, String Email, String Username)
AddHealthCondition	This is the control that adds user's health condition to the system. It extends MaintainIntakeRecords. *Responsibilities*: public bool isEmpty(String healthcondition) public void AddHealthCondition(int healthconditionID) public void AddHealthCondition(String healthcondition) public void saveHealthCondition(String healthcondition) public void editHealthCondition(int healthconditionID) public void deleteHealthCondition(int healthconditionID)

viewUserProfile	This is the control that lets the user view his/her user profile details. *Responsibilities*: public void viewUserProfile(int userID)
viewCurrentIntake	This is the control that lets the user view a food or water intake. *Responsibilities*: public void viewIntake(int waterintakeid) public void viewIntake(int foodintakeid)
LogInManager	This is the control that checks the credentials of the user log-in whether it is authenticated or not. *Responsibilities*: public void CheckCredentials(String username, String password)

Data Access Objects Packages:

DAO Name	**Description**

UserDAO	This data access object is responsible for getting user profile information data from a file referenced with filename, UserData.csv
	
	Methods:
	
	private void accessUserDataFile();
	
	private User getUser(int userid);
	
	private User[] getUserList();
	
	private void addUser(User newuser, int userid);
	
	private void editUser(User newuser, int userid);

IntakeDAO	This data access object is responsible for getting user intake information data from a file referenced with filename, UserIntake.csv
	
	Methods:
	
	private void accessUserIntake();
	
	private Intake[] getIntakeList(int userid);
	
	private Intake getIntake(int userid, int waterintakeID);
	
	private Intake getIntake(int userid, int foodintakeID);
	
	private void addIntake(Intake newintake, int userid);
	
	private void editIntake(Intake newintake, int userid);

Transfer Objects Package:

Class Name	Description

User	This is the entity class user, which contains the data about the user.
	Attributes:
	private int userid
	public String username
	private String password
	Methods:
	private void setUserID(int userID);
	private void setUsername(String username);
	private void setPassword(String password);
	…
	private int getUserID();
	private String getUsername();
	private String getPassword();

Intake	This is the entity class intake, which contains the data about the intakes.
	Attributes:
	private int waterintakeid
	private int foodnameid
	private String watercontainer
	private String foodname
	private float amount
	private float calorie
	private Date date_consumed
	private Date time_consumed
	Methods:
	private void setWaterIntakeID(int waterIntakeID);
	private void setFoodnameID(int foodNameID);
	private void setWaterContainer(String watercontainer);
	private void setFoodName(String foodname);
	private void setAmount(float amount);
	private void setCalorie(float calorie);
	private void setDateConsumed(Date date_consumed);

	private void setTimeConsumed(Date time_consumed)
	…
	private void getWaterIntakeID();
	private void getFoodnameID();
	private void getWaterContainer();
	private void getFoodName();
	private void getAmount();
	private void getCalorie();
	private void getDateConsumed();
	private void getTimeConsumed();

User Profile	This is the entity class user profile, which contains the data about the user profiles.
	Attributes:
	private int userID
	private float weightInKilograms
	private float heightInCentimeters
	private String ContactNumber
	private String Email
	private String Username
	private Date Birthdate
	private int healthConditionID
	private String healthCondition
	Methods:
	private void setUserID(int userID);
	private void setWeightInKilograms(float weight);
	private void setHeightInKilograms(float height);
	private void setContactNumber(String contactNumber);
	private void setEmail(String email);
	private void setUsername(String username);
	private void setBirthdate(Date birthdate);
	private void setHealthConditionID(int healthConditionID);
	private void setHealthCondition(String healthCondition);
	…

	private void getUserID();
	private void getWeightInKilograms();
	private void getHeightInKilograms();
	private void getContactNumber();
	private void getEmail();
	private void getUsername();
	private void getBirthdate();
	private void getHealthConditionID();
	private void getHealthCondition();

Data Sources Package:

File Name or Database Name	Description
UserData.csv	This is the data source for the user, which contains the data about the user (userid, username, password) and the users' profile.
UserIntake.csv	This is the data source of intake records found in the health and fitness record management system.