**CS673F13 Software Engineering**

**Group Project - Type II Diabetes Management CDSS**

**Software Design Document**



|  |  |  |  |
| --- | --- | --- | --- |
| Team Member | Role(s) | Signature | Date |
| Bogdan Chayka | Backup project leader, Design leader | *Bogdan Chayka* | 11/07/2013 |
| Jason Lu | Configuration leader |  | 11/07/2013 |
| Jeff Andre | Project leader, Requirements | *Jeff Andre* | 11/07/2013 |
| Maura Huff | Implementation leader |  | 11/07/2013 |
| Wenjie Shi(Jenny) | Environment and integration leader |  | 11/07/2013 |
| Yike Xue (Eva) | QA leader |  | 11/07/2013 |
| Yingyuan (Allen) Zhang | Requirements leader |  | 11/07/2013 |
| Yulu Liu | Design leader |  | 11/07/2013 |

**Revision history**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Author** | **Date** | **Change** |
| **1.0** | **Jeff Andre, Bogdan Chayka** | 11/07/2013 |  |
|  |  |  |  |

[Introduction](#h.87t9hln2vjz0)

[Software Architecture](#h.buttcq9i221r)

[Design Patterns](#h.x18fj36s1121)

[Key Algorithms](#h.mtfbusfb0eq3)

[Classes and Methods](#h.7ucksmkf6rzx)

[References](#h.15tmymhipvdv)

[Glossary](#h.8n34lvocupub)

# Introduction

In this section, give an overview of this document, and also address the design goals of your software system.

This web application is designed to be a teaching tool that demonstrate implementation of a real clinical algorithm (part of [AACE Diabetes Management Algorithm 2013](https://www.aace.com/files/aace_algorithm.pdf)), examples of successful treatment decisions, situations where clinical algorithms may fail, and to provide a platform for familiarization of students with the process of development and implementation and use of medical algorithms.

This application is divided into five major parts:

a) user interface

b) a database

c) a type II diabetes management algorithm

d) health calculators

e) security

Figure 1 shows a diagram of the application.

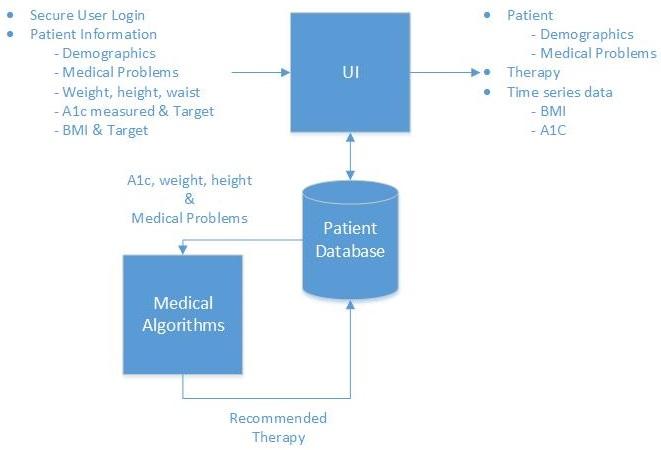


Figure 1: Diagram of the application

# Software Architecture

In this section, you will describe the decomposition of your software system, which include each component (which may be in terms of package or folder) and the relationship between components. You shall have a diagram to show the whole architecture, and class diagram for each component. The interface of each component and dependency between components should also be described. If any framework is used, it shall be defined here too. Database design should also be described if used.

Model View Controller architecture is used for development of the application.

**1. Model**

1.1 Database

MySQL is used to implement the database and CakePHP naming conventions to name tables.

Figure 2 shows a database schema.

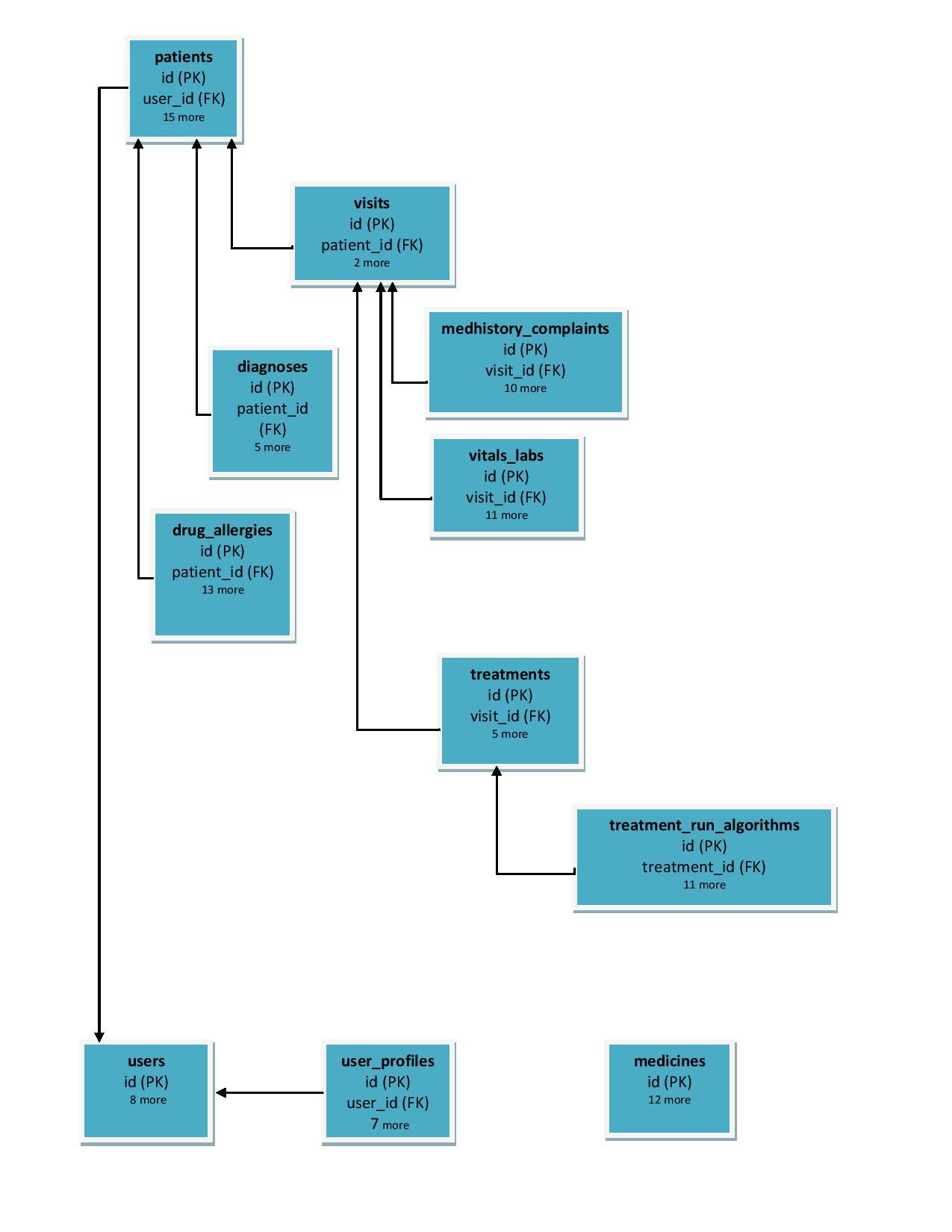


Figure 2: Database schema

Detailed descriptions of tables:

patients

|  |  |  |  |
| --- | --- | --- | --- |
| **attributes** | **types** | **constraints** | **keys** |
| id | char (36) | not null | PK |
| created | datetime | not null |  |
| modified | datetime | not null |  |
| user\_id | int | not null | FK |
| patient\_number | varchar(8) | not null |  |
| patient\_firstname | varchar(255) | not null |  |
| patient\_lastname | varchar(255) | not null |  |
| patient\_middlename | varchar(255) |  |  |
| dob | date | not null |  |
| picture | blob |  |  |
| occupation | varchar(255) |  |  |
| gender | enum | not null  ‘Male’, ‘Female’ |  |
| race | enum | not null  “African or African American”,  “Asian or Asian American”,  “Caucasian or European American”,  “Native American or Native Alaskan ”,  “Other Race” |  |
| street | varchar(255) |  |  |
| postal\_code | char(5) |  |  |
| city | varchar(255) |  |  |
| state | enum | All states  (see notes) |  |

visits

|  |  |  |  |
| --- | --- | --- | --- |
| **attributes** | **types** | **constraints** | **keys** |
| id | int | not null | PK |
| created | datetime | not null |  |
| modified | datetime | not null |  |
| patient\_id | char(36) | not null | FK |

medhistory\_complaints

|  |  |  |  |
| --- | --- | --- | --- |
| **attributes** | **types** | **constraints** | **keys** |
| id | int | not null | PK |
| created | datetime | not null |  |
| modified | datetime | not null |  |
| visit\_id | int | not null | FK |
| complaints | text | not null |  |
| hypo | enum | not null  “yes” or “no” |  |
| weight\_gain | enum | not null  “yes” or “no” |  |
| renal\_gu | enum | not null  “yes” or “no” |  |
| gi\_sx | enum | not null  “yes” or “no” |  |
| chf | enum | not null  “yes” or “no” |  |
| cvd | enum | not null  “yes” or “no” |  |
| bone | enum | not null  “yes” or “no” |  |

vitals\_labs

|  |  |  |  |
| --- | --- | --- | --- |
| **attributes** | **types** | **constraints** | **keys** |
| id | int | not null | PK |
| created | datetime | not null |  |
| modified | datetime | not null |  |
| visit\_id | int | not null | FK |
| weight | float | not null |  |
| height | float | not null |  |
| bps | int | not null |  |
| bpd | int | not null |  |
| bmi | float |  |  |
| bmi\_status | varchar(255) |  |  |
| A1c | float | not null |  |
| eGFR | int |  |  |
| notes | text |  |  |

diagnoses

|  |  |  |  |
| --- | --- | --- | --- |
| **attributes** | **types** | **constraints** | **keys** |
| id | int | not null | PK |
| created | datetime | not null |  |
| modified | datetime | not null |  |
| patient\_id | char(36) | not null | FK |
| dxname | enum | ‘Non-insulin-dependent diabetes mellitus’ |  |
| icd10code | enum | not null,  “E11” |  |
| Icd9code | enum | Not null,  “250.00”, “250.02” |  |

drug\_allergies

|  |  |  |  |
| --- | --- | --- | --- |
| **attributes** | **types** | **constraints** | **keys** |
| id | int | not null | PK |
| created | datetime | not null |  |
| modified | datetime | not null |  |
| patient\_id | char(36) | not null | FK |
| met | enum | not null,  ‘yes’,’no’ |  |
| dpp\_4i | enum | not null,  ‘yes’,’no’ |  |
| glp\_1ra | enum | not null,  ‘yes’,’no’ |  |
| tzd | enum | not null,  ‘yes’,’no’ |  |
| agi | enum | not null,  ‘yes’,’no’ |  |
| colsvl | enum | not null,  ‘yes’,’no’ |  |
| bcr\_or | enum | not null,  ‘yes’,’no’ |  |
| su\_gln | enum | not null,  ‘yes’,’no’ |  |
| insulin | enum | not null,  ‘yes’,’no’ |  |
| sglt\_2 | enum | not null,  ‘yes’,’no’ |  |
| praml | enum | not null,  ‘yes’,’no’ |  |

treatments

|  |  |  |  |
| --- | --- | --- | --- |
| **attributes** | **types** | **constraints** | **keys** |
| id | int | not null | PK |
| created | datetime | not null |  |
| modified | datetime | not null |  |
| visit\_id | int | not null | FK |
| prescriber\_username | varchar(255) | not null |  |
| a1c\_goal | float |  |  |
| weight\_goal | float |  |  |

treatment\_run\_algorithms

|  |  |  |  |
| --- | --- | --- | --- |
| **attributes** | **types** | **constraints** | **keys** |
| id | int | not null | PK |
| created | datetime | not null |  |
| modified | datetime | not null |  |
| treatment\_id | int | not null | FK |
| type | enum | not null  “lifestyle modification”  “monotherapy”,  “dual\_therapy”,  “triple\_therapy” |  |
| recommendations | text |  |  |
| medicine\_name\_one | varchar(255) |  |  |
| dose\_one | float |  |  |
| medicine\_name\_two | varchar(255) |  |  |
| dose\_two | float |  |  |
| medicine\_name\_three | varchar(255) |  |  |
| dose\_three | float |  |  |
| edited\_by\_user | enum | not null “yes”, ”no” |  |

medicines

|  |  |  |  |
| --- | --- | --- | --- |
| **attributes** | **types** | **constraints** | **keys** |
| id | int | not null | PK |
| created | datetime | not null |  |
| modified | datetime | not null |  |
| medicine\_name | varchar(255) | not null |  |
| min\_dose | float | not null |  |
| max\_dose | float | not null |  |
| metric | char(8) | not null |  |
| hypo | enum | “0”,”1”,”2”,”3” |  |
| weight | enum | “0”,”1”,”2”,”3” |  |
| renal\_gu | enum | “0”,”1”,”2”,”3” |  |
| gi\_sx | enum | “0”,”1”,”2”,”3” |  |
| chf | enum | “0”,”1”,”2”,”3” |  |
| cvd | enum | “0”,”1”,”2”,”3” |  |
| bone | enum | “0”,”1”,”2”,”3” |  |

users

|  |  |  |  |
| --- | --- | --- | --- |
| **attributes** | **types** | **constraints** | **keys** |
| id | int | not null | PK |
| created | datetime | not null |  |
| modified | datetime | not null |  |
| email | varchar(128) | not null |  |
| password | char(64) | not null |  |
| reset\_token | char(64) |  |  |
| username | varchar(45) |  |  |
| openid | varchar(64) |  |  |
| service | varchar(45) |  |  |
| activated | tinyint | not null  default 0 |  |
| role | int |  |  |

user\_profiles

|  |  |  |  |
| --- | --- | --- | --- |
| **attributes** | **types** | **constraints** | **keys** |
| id | int | not null | PK |
| created | datetime | not null |  |
| modified | datetime | not null |  |
| user\_id | int | not null | FK |
| title | varchar(255) |  |  |
| firstname | varchar(255) | not null |  |
| lastname | varchar(255) | not null |  |
| phone | varchar(255) |  |  |
| address | varchar(255) |  |  |

1.2 Model

users

patients

diagnoses

drug\_allergies

visits

medhistory\_complaints

vitals\_labs

treatments

treatments\_medicines

**Glossary for database**:

Hypo – hypoglycemia

Weight – weight gain

Renal/GU Genitourinary and Renal symptoms

GI Sx – Gastrointestinal symptoms

CHF – coronary heart disease ( ischemic heart disease IHD)

CVD – cardiovascular disease

BONE – osteoporosis

eGFR – estimated glomerular filtration rate

a1c - blood glucose test

**Notes:**

Meaning of enum In medicines table

“0” - neutral

“1” - few adverse events or possible benefits

“2” – use with caution

“3” – likelihood of adverse effects

States in patients table

'Alabama','Alaska','Arizona','Arkansas','California','Colorado','Connecticut','Delaware','Florida','Georgia','Hawaii','Idaho','Illinois','Indiana','Iowa','Kansas','Kentucky','Louisiana','Maine','Maryland','Massachusetts','Michigan','Minnesota','Mississippi','Missouri','Montana','Nebraska','Nevada','New Hampshire','New Jersey','New Mexico','New York','North Carolina','North Dakota','Ohio','Oklahoma','Oregon','Pennsylvania','Rhode Island','South Carolina','South Dakota','Tennessee','Texas','Utah','Vermont','Virginia','Washington','West Virginia','Wisconsin','Wyoming'

2. View

Users:

edit

index

signup

profile

password

Patients:

search

add

edit

show

index

Visits:

add

current

show

edit

gcalgorithm

index

Medicines:

add

edit

view all

view

3. Controller

usersController:

login()

logout()

add()

edit()

delete()

show()

sign\_up()

password()

activate()

deactivate()

patientsController:

search()

add()

edit()

show()

delete()

set\_patient\_id()

visitsController:

index()

add()

\_add()

current()

show()

gcalgorithm()

edit()

medicinesController:

add()

index()

edit()

delete()

show()

# Design Patterns

In this section, you shall describe any design patterns used in your software system.

* Agile Development

Generally speaking, our project is applied to the Agile Development technique with Iterative and Incremental Model to develop our project. The life cycle is:

1. Obtain requirements for next period’s segment
2. Refactor to accommodate new requirements
3. Modify code and test code base to handle additional requirements
4. Refactor to clean up
5. (Finish one iteration)Obtain high level requirements and start from step1.

This technique motivates developers and satisfy the customer (we can see the prototype and basic pages of the website early); The software is thoroughly tested early as soon as some modules are integrated and always demonstrated to the customer as well as other developers even some functions have not been achieved yet.

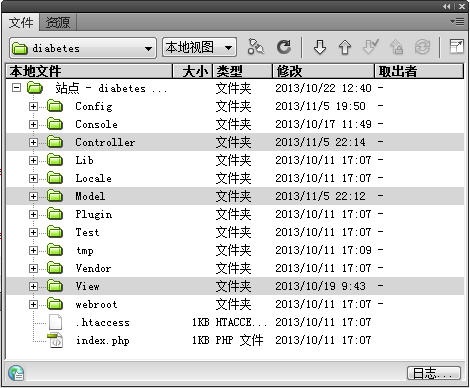
* MVC structure

MVC architecture separates the business logic and application data from the presentation data to the user. It consists of three parts: Model, View and Controller.

The model represents the business or domain logic of the application, typically with data backed by a database.

The view is what shows to the users. It is selected by the controller and renders the appropriate user interface.

The model is the core of this architecture. It links the model and view. The controller locates the appropriate action method in the controller, gets values to use as the action method's arguments, and handles any errors that might occur when the action method runs. It then renders the requested view.



* Test Driven Development

After the first iteration, we gradually add some TDD concept into our development process. For example, members would test some special cases and make them fail, then we try to fix these bugs and make the test cases passed. According to the test results, we can find out where we should improve and modify. clearly After the tests, we would modify our code until the tests pass. That is how our software getting improved.

The workflow is as following:

1. Create a test case based on the requirements and make it failed.

2. Write code to pass the test in Step1.

3. Refactor so that the code is made to be readable, maintainable and make sure it can still pass the test.

4. Repeat for each new test until all the requirements are met.

# Key Algorithms

A key algorithm used in this project is a part of the AACE Diabetes Management Algorithm 20131 which focuses on glycemic control of patients with type 2 diabetes mellitus. The two figures below show flow diagrams of the algorithm. Figure 1 shows the overall flow diagram for therapy selection based on A1c levels, and figure 2 shows the flow diagram for medication selection for each therapy. The algorithm primarily recommends a therapy for type 2 diabetes patients which includes therapeutic lifestyle changes plus one, two, or three medications - referred to as monotherapy, dual therapy and triple therapy respectively.

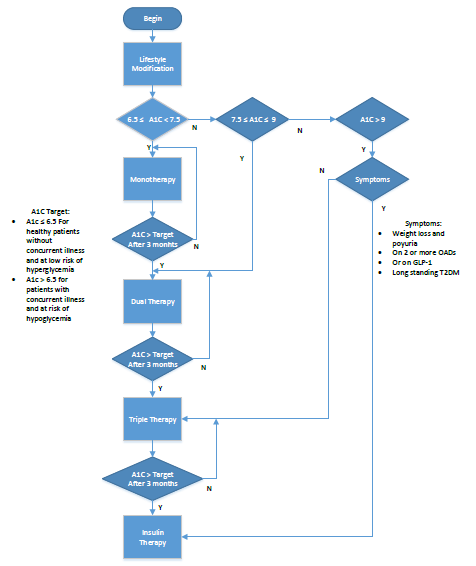


Figure 1. Glycemic Control Algorithm Flow Diagram

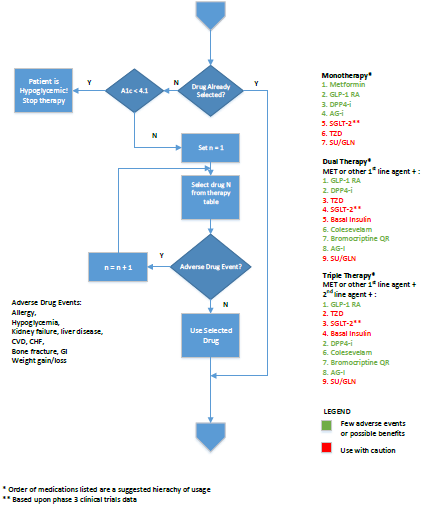


Figure 2. Monotherapy, Dual Therapy, & Triple Therapy flow diagram

# Classes and Methods

This part can be a reference to automatic generated document for all classes and methods.

**Classes:**

usersController, patientsController, visitsController, medicinesController

**Methods:**

usersController:

login()

logout()

add()

edit()

delete()

show()

sign\_up()

password()

activate()

deactivate()

patientsController:

search()

add()

edit()

show()

delete()

set\_patient\_id()

visitsController:

index()

add()

\_add()

current()

show()

gcalgorithm()

edit()

medicinesController:

add()

index()

edit()

delete()

show()

# References

[1] AACE Comprehensive Diabetes Management, *Endocr Pract.* 2013;19(Suppl 2).

# Glossary

a1c - blood glucose test

AACE - American Association of Clinical Endocrinologists

A1c - Common blood test used to diagnose and manage type 1 and type 2 diabetes.

Also, known as glycated hemoglobin, glycosylated hemoglobin, hemoglobin A1C and

HbA1c.

BONE – osteoporosis

CHF – coronary heart disease ( ischemic heart disease IHD)

CVD – cardiovascular disease

eGFR – estimated glomerular filtration rate

GI Sx – Gastrointestinal symptoms

Hypo – hypoglycemia

Renal/GU Genitourinary and Renal symptoms

Weight – weight gain