

CSCC01 Sprint Backlog #1

Team Name: Ctrl-Alt-Elite
Team Number: 11
Date: October 29th, 2018

Table of Contents

Tasks	3
Design Framework	4
Sprint Plan	5
Sprint Report	6
Burndown chart	7

Tasks

U1:T1 - Create Database Structure: Given the main requirements of this new system, we need to store certain type of data besides the iCare templates. In this task, the main entities for user accounts and templates need to be created, as well as initializing the attributes for each.

U1:T2 - Create User Interface: Create the user interface for when a user needs to create a new entity/table for a template.

U1:T3 - Create Base classes: Create base classes for templates and users involved in adding templates. Implement interfaces for the MVP Pattern.

U1:T4 - Create API for Databases: Create API for the SQL Database.

U1:T5 - Integrating User Interface and Database API: Connect classes and functionalities from the User Interface to the API

U1:T6 - Create Database Templates: Implement “Client Profile” and “Needs Assessment and Referrals Service” templates in the Database.

U1:T7 - Create Database Templates: Implement “Community Connections” and “Language Training - Client Enrolment” templates in the Database.

U1:T8 - Create Database Templates: Implement “Employment Related Services” and “Information and Orientation” templates in the Database.

U1:T9 - Create Database Templates: Implement “Language Training - Course Setup” and “Language Training - Client Exit” templates in the Database.

U1:T10 - Database Integrations: Integration of T6 to T9 by merging the different database pieces into one.

U1:T11 - Parse Column names: Get raw values from an Excel file, return the name of the template and the column names.

U2:T1 - Update Database: Set up connection between our application and the database, to be able to store the data that has been inputted.

U2:T2 - Create User Interface: Create the user interface for agencies to select an excel file to upload to send data through our application.

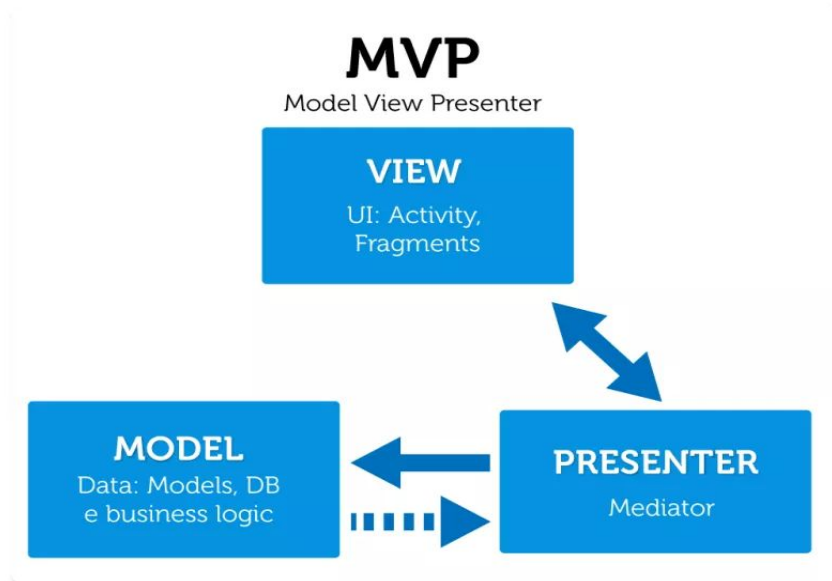
Legend:

Bold - Completed in this sprint.

Grey - Completed on the previous sprints.

Italics - New tasks that were created during Scrum meetings.

Design Framework



For this project, we will use Model View Presenter (MVP) design pattern to divide the application in three major aspects: View, Presenter and Model.

- View will directly interact with the user and does not contain any logic.
- Model represents the classes that describes the business logic and business rules on how the data can be manipulated. Our database and database API will be in model.
- Presenter receives input for View, process the user request with Model and then return results back through View. For instance, Presenter will be in charge of getting template names and uploading data for an specific template.

Sprint Plan

User Stories	Tasks	Dependency	Story Points	1	2	3	4	5	6
U1	U1:T1		2	J:1	J:1				
U1	U1:T2		7		T:1	T:1	T:1	T:1	T:1
U1	U1:T3		2					V:1	V:1
U1	U1:T4	U1:T1	4						
U1	U1:T5	U1:T2, U1:T4	2						
U1	U1:T6		3			J:1	J:1	J:1	
U1	U1:T7		3		V:1	V:1	V:1		
U1	U1:T8		3		A:1	A:1	A:1		
U1	U1:T9		3		L:1	L:1	L:1		
U1	U1:T10		2						J:1
U1	U1:T11		6					L:1	L:1
U2	U2:T1		4						
U2	U2:T2		4					A:1	A:1

Sprint Report

User Stories	Tasks	Dependency	Story Points	1	2	3	4	5	6
U1	U1:T1		2	J:1	J:1				
U1	U1:T2		7			T:1	T:1	T:1	T:1
U1	U1:T3		2						
U1	U1:T4	U1:T1	4						
U1	U1:T5	U1:T2, U1:T4	2						
U1	U1:T6		3			J:1	J:1	J:1	J:1
U1	U1:T7		3			V:1	V:1	V:1	V:1
U1	U1:T8		3			A:1	A:1	A:1	A:1
U1	U1:T9		3			L:1	L:1	L:1	L:1

Burndown chart

Days	0	1	2	3	4	5	6
Provisional (in story points)	30	30	28	28	19	16	12
Actual (in story points)	30	30	28	28	28	28	16

