

POLITECHNIKA WARSZAWSKA
WYDZIAŁ ELEKTRYCZNY
INSTYTUT ELEKTROTECHNIKI TEORETYCZNEJ
I SYSTEMÓW INFORMACYJNO-POMIAROWYCH
PRACA DYPLOMOWA MAGISTERSKA
Na kierunku **INFORMATYKA**



Krzysztof Kozłowski
Nr indeksu 188140

Rok akademicki: 2006/2007
Warszawa, 26 listopada 2007

System obsługi zawodów sportowych agility

Zakres pracy:

1. Analiza potrzeb użytkownika
2. Zaprojektowanie systemu
3. Wykonanie dwóch aplikacji wchodzących w skład systemu
4. Dokumentacja

*(Podpis i pieczęć
Kierownika Zakładu
Dydaktycznego)*

Opiekun naukowy: dr inż. Jacek Korytkowski
Recenzent: prof. dr hab. Kazimierz Mikołajuk

Termin wykonania: 26 listopada 2007

Spis treści

1	Aplikacja AgileWeb	1
1.1	Instalacja	1
1.1.1	Wymagania	1
1.1.2	Instalacja plików	2
1.1.3	Baza danych	3
1.2	Instrukcja użytkownika	3
1.2.1	Rozpoczęcie pracy – konto	3
1.2.2	Udział w zawodach	5
1.2.3	Zarządzanie zawodami (administrator)	6
1.2.4	Lista startowa (administrator)	8
1.2.5	Wyniki zawodów (administrator)	9
2	Aplikacja AgileDesk	10
2.1	Instalacja	10
2.1.1	Wymagania	10
2.1.2	Instalacja wymaganych komponentów na GNU/Linux	11
2.1.3	Instalacja wymaganych komponentów na Microsoft Windows	11
2.1.4	Konfiguracja AgileDesk	12
2.2	Instrukcja użytkownika	12
2.2.1	Uruchomienie	12
2.2.2	Rozpoczęcie pracy	12
2.2.3	Wyświetlenie i modyfikacja listy startowej	13
2.2.4	Wprowadzanie wyników biegów	14
2.2.5	Wyniki zawodów i eksport	15
	Bibliografia	16

Rozdział 1

Aplikacja AgileWeb

1.1 Instalacja

1.1.1 Wymagania

Jak wymienione zostało to w części *Ograniczenia środowiskowe* projektu AgileWeb (ograniczenie *EC-1* w tabeli ??), aplikacja instalowana musi być na serwerze WWW z obsługą PHP w wersji 5 lub wyższej. Serwerem WWW może być Apache [21] na platformie GNU/Linux lub Unix, ale równie dobrze wykorzystać można inny serwer wspierający używanie PHP (poprzez moduły współdzielone lub CGI/FastCGI).

AgileWeb bezpośrednio nie jest przywiązane do konkretnego DBMS (Database Management System), bo za obsługę bazy danych odpowiada framework CodeIgniter. Jednakże do poprawnej pracy DBMS musi obsługiwać transakcje, jak wymienione zostało w wymaganiu *NFR-1* z *Wymagań pozafunkcjonalnych* projektu (tabela ??). Przykładami takich DBMS są:

- PostgreSQL [22];
- MySQL w wersji 5 lub nowszej (z użyciem tabel InnoDB) [20].

Dostęp do aplikacji AgileWeb odbywa się poprzez przeglądarkę WWW. Przeglądarka powinna obsługiwać XHTML 1.0 [23] oraz CSS 1.0 [24] oraz obiekt XMLHttpRequest [25] (poprzez język JavaScript). Powyższe wymagania spełniają nowe wersje przeglądarek: Opera, Mozilla Firefox, Konqueror, Safari oraz do pewnego stopnia Microsoft Internet Explorer w wersji 6.0 i 7.0. Na tych przeglądarkach aplikacja była testowana i sprawowała się poprawnie.

1.1.2 Instalacja plików

Proces instalacji zostanie zobrazowany na przykładzie uczelnianego serwera volt. Docelowym miejscem umiejscowienia plików będzie katalog `/home/stud/kozlowk3/WWW/agileweb/` dalej oznaczany przez `~/`, widoczny w internecie jako `http://iem.pw.edu.pl/~kozlowk3/agileweb`.

Pierwszym etapem instalacji AgileWeb jest wgranie plików CodeIgnitera. Jego instalacja jest bardzo prosta – szczegółowo opisuje ją też podręcznik użytkownika [7].

1. Pobranie ze strony CodeIgnitera [5] aktualnej paczki z frameworkiem.
2. Rozpakowanie archiwum do katalogu na serwerze WWW, w którym będzie umieszczona cała aplikacja AgileWeb:

```
unzip CodeIgniter_1.5.4.zip
mv CodeIgniter_1.5.4/* ~/
```

3. Edycja pliku `~/system/application/config/config.php` i ustawienie w nim:
 - `$config['base_url']` – na adres URL do aplikacji, czyli miejsca z plikiem `index.php` (z końcowym slashem), np.: `http://iem.pw.edu.pl/~kozlowk3/agileweb/`;
 - `$config['sess_use_database']` – na wartość `TRUE`;
 - `$config['sess_match_ip']` – na wartość `TRUE`;
 - `$config['cookie_domain']` – na nazwę domeny, w której mają być tworzone ciasteczka (opcjonalne), np. `iem.pw.edu.pl`;
 - `$config['cookie_path']` – na ścieżkę w adresie URL dla ustawianych ciasteczek (opcjonalne), np. `/~kozlowk3/agileweb/`;
4. Edycja pliku `~/system/application/config/agilewebconstants.php` i ustawienie w nim:
 - `FROM_MAIL` – na adres email, z którego mają być wysyłane maile, np.: `webmaster@iem.pw.edu.pl`;
 - `WEBSITE_ADMIN` – na dane kontaktowe do administratora;
5. Edycja pliku `~/system/application/config/database.php` i ustawienie w nim połączenia z bazą danych:
 - `$db['default']['hostname']` – nazwa hosta bazy danych (np. `localhost`);

- `$db['default']['username']` – nazwa użytkownika;
- `$db['default']['password']` – hasło;
- `$db['default']['database']` – nazwa bazy danych;
- `$db['default']['dbprefix']` – prefiks nazwy każdej z tabel, skrypt instalacyjny SQL dołączony do aplikacji AgileWeb wymaga by ten prefiks brzmiał `agileweb_` (zwróć uwagę na końcowe podkreślenie);

Kolejnym etapem jest wgranie biblioteki XAJAX w wersji 0.2.x. Z pobranego ze strony domowej projektu [9] archiwum należy wypakować pliki i:

1. katalog `xajax_js` wraz z zawartością umieścić w korzeniu aplikacji AgileWeb (tj. obok pliku `index.php` z CodeIgniter);
2. plik `xajax.inc.php` umieścić w `~/system/application/libraries` zmieniając nazwę na `xajax.php`;
3. pliki `xajaxCompress.php` oraz `xajaxResponse.inc.php` umieścić w `~/system/application/libraries` (bez zmiany nazwy);

1.1.3 Baza danych

Obok kodu źródłowego dołączonego na nośniku do pracy znaleźć można plik SQL `agileweb_install.sql` instalujący wymagane tabele w bazie danych. Plik przeznaczony jest dla MySQL i wymaga wsparcia dla tabel InnoDB. Przykładowo korzystając z konsolowego klienta `mysql` polecenie utworzenia wymaganej struktury bazy danych może wyglądać następująco:

```
mysql -p -u nazwa_uzytkownika \  
    nazwa_bazy_danych < agileweb_install.sql
```

Skrypt `agileweb_install.sql` utworzy tabele w bazie o nazwach zaczynających się od `agileweb_`, stąd ważne jest ustawienie zmiennej `dbprefix` w konfiguracji dostępu do bazy danych w CodeIgniter. Skrypt doda też konto podstawowego administratora systemu – użytkownika `admin` z hasłem `admin`.

1.2 Instrukcja użytkownika

1.2.1 Rozpoczęcie pracy – konto

Użytkownik niezalogowany ma ograniczony dostęp do aplikacji. Może tylko ujrzeć listę zawodów oraz ich wyniki lub listy startowe. Dalsza funkcjonalność zostaje udostępniona po utworzeniu konta i zalogowaniu się na nie.

[Strona główna](#) / utworzenie nowego konta

Utworzenie nowego konta

Pola obowiązkowe:

Nazwa użytkownika:	<input type="text"/>
Hasło:	<input type="password"/>
Powtórz hasło:	<input type="password"/>
Adres e-mail:	<input type="text"/>
Imię:	<input type="text"/>
Nazwisko:	<input type="text"/>

Pola nieobowiązkowe:

Telefon:	<input type="text"/>
Ulica:	<input type="text"/> (ulica, numer domu, numer mieszkania)
Kod pocztowy:	<input type="text"/>
Miasto:	<input type="text"/>
Junior:	<input type="checkbox"/> (nie ukończył 18 lat do dnia zawodów)
Debiutant:	<input type="checkbox"/>
Nazwa klubu:	<input type="text"/>

Rysunek 1.1: AgileWeb – ekran rejestracji w systemie

Po wypełnieniu formularza rejestracyjnego (rys. 1.1) na podany adres e-mail zostanie wysłany list potwierdzający założenie konta. W liście znajdować się będzie adres URL, po którego odwiedzinach konto zostanie aktywowane. Do czasu aktywacji konto ma te same uprawnienia, co użytkownik niezalogowany – gość. Użytkownik identyfikowany jest w systemie poprzez nazwę konta oraz hasło.

Jeśli użytkownik zapomni hasło do swojego konta, może skorzystać z opcji jego przypomnienia. Będzie jednakże musiał podać prawidłową nazwę konta i adres e-mail z nim związany. Jeśli obydwie dane poda poprawne, to na ten adres zostanie wysłane nowe, tymczasowe hasło, na które będzie mógł się zalogować. Do czasu użycia tego tymczasowego hasła, jego główne nie zostanie zmienione. Po zalogowaniu się użytkownik otrzymuje dostęp do zarządzania swoim kontem w ramach aplikacji (rys. 1.2). Poza zmianą danych osobowych, hasła czy adresu e-mail powinien dodać w nim psy, z którymi zamierza startować w zawodach.

AgileWeb

[Strona główna](#) [O aplikacji/Pomoc](#) [Zawody](#) [Konto](#) [Wyloguj się](#)

[Strona główna](#) / [test test](#)

test test - konto

Nazwa użytkownika: **test test**
 Adres e-mail: **tester**
 Junior: **kozik1@o2.pl**
 Debiutant: **nie**
 Telefon: **nie**
 Adres:
 Nazwa klubu:

- ◆ [Modyfikacja konta](#)
- ◆ [Dodaj nowego psa](#)

Twoje psy

Imię	Rasa	Płeć	Wysokość	Data urodzenia	Imię rodowodowe	Weteran
Reks	jamnik	pies	20 cm	2002-10-10		zmień / usuń
Azor	Kundel	pies	40 cm	2000-05-11		tak: zmień / usuń

Rysunek 1.2: AgileWeb – ekran konta użytkownika

1.2.2 Udział w zawodach

Posiadając konto w systemie użytkownik może zgłosić udział w zawodach do dniach ich rozpoczęcia. Przy każdym czynnych zawodach widnieć będzie odnośnik *zgłoś uczestnictwo* prowadzący do formularza dodania lub modyfikacji swojego zgłoszenia udziału.

[Strona główna](#) / [zawody - testowe w test](#) / [modyfikacja zgłoszenia](#)

Zgłoszenie do zawodów testowe w test

Zgłoszenie nr 1:

Pies:

Konkurencja:

Kategoria wzrostowa:

[Usuń to zgłoszenie](#)

Rysunek 1.3: AgileWeb – ekran modyfikacji zgłoszenia w zawodach

Ekran z formularzem (rys. 1.3) zawiera wszystkie zgłoszenia danego użytkownika w tych zawodach. Przy istniejących zgłoszeniach umieszczony jest odnośnik *Usuń to zgłoszenie* w celu jego anulowania. W każdym zgłoszeniu

użytkownik musi wybrać jednego ze swoich psów, konkurencję i kategorię wzrostową. Dostępne dla danego psa konkurencje oraz kategorie wzrostowe są uzależnione od samej konfiguracji zawodów ustalanej przez administratora oraz ewentualnie od wcześniejszych zgłoszeń w tych zawodach. Aplikacja, poprzez AJAX, reaguje na wybór użytkownika bez konieczności ponownego ładowania formularza:

- po wyborze psa zostanie uaktualniona lista wyboru konkurencji (bazując na ewentualnych wykluczeniach pomiędzy jednoczesnym udziałem w kilku konkurencjach);
- po wyborze konkurencji zostanie uaktualniona lista dostępnych kategorii wzrostowych w tej konkurencji oraz pojawi się nazwa sugerowanej kategorii wzrostowej bazując na wysokości danego psa.

Po uzupełnieniu pól formularza (psa, konkurencji i kategorii wzrostowej), aby zgłoszenie zostało zachowane, użytkownik powinien nacisnąć *Zapisz zmiany*. Użytkownik może modyfikować (w tym i usuwać) zgłoszenia w zawodach do czasu ich rozpoczęcia.

1.2.3 Zarządzanie zawodami (administrator)

Zarządzanie zawodami możliwe jest tylko z konta o uprawnieniach administratora. Wszystkie elementy dalej opisane znaleźć można pod zakładką *Zawody* z menu aplikacji.

[Strona główna](#) / [lista zawodów](#) / [zmiana konkurencji](#)

Zmiana konkurencji

Zmień konkurencje:

Nazwa konkurencji	Opis	Kolejność	Dług. toru	Połącz z	
Agility		1	0	Jumping	usuń
Jumping		2	0	Agility	usuń
Youngsters	Dla młodych psów	3	0	brak	usuń

Dodaj nową konkurencję:

Nazwa konkurencji	Opis	Kolejność	Dług. toru	Połącz z
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	brak <input type="text"/>

Rysunek 1.4: AgileWeb – ekran modyfikacji konkurencji

Administrator powinien wpięrow dodać odpowiednią ilość konkurencji (rys 1.4), które będą przypisywane poszczególnym zawodom. Samo przypisanie

konkurencji do zawodów odbywa się przy dodawaniu nowych lub modyfikacji istniejących zawodów. Tworzenie konkurencji łącznych (będących zasadniczo tylko częścią wspólną wyników biegów z obydwu) odbywa się poprzez ustawienie pola *Połącz z* na nazwę drugiej konkurencji. Jedna konkurencja łączona może składać się tylko z dwóch konkurencji.

Administrator może ustawić wykluczenia pomiędzy konkurencjami. Wykluczenie polega na zabronieniu zawodnikowi startu w danej konkurencji, jeśli już zgłosił swoje uczestnictwo do innej (wykluczającej).

The screenshot shows a web interface titled "Łączenie kategorii wzrostowych i konkurencji". At the top, there is a breadcrumb trail: "Strona główna / lista zawodów / łączenie kategorii wzrostowych i konkurencji". The main content is divided into two sections:

- Obecne połączenia:** This section contains two columns of dropdown menus. The left column is labeled "Konkurencja" and has three dropdowns, all set to "Agility". The right column is labeled "Kategoria wzrostowa" and has three dropdowns set to "Small", "Medium", and "Large". To the right of each dropdown is a blue "usuń" (delete) link. Below these columns are two buttons: "Wprowadź zmiany" and "Anuluj/Zakończ".
- Dodaj nowe połączenie:** This section contains two dropdown menus. The left one is labeled "Konkurencja" and is set to "Agility". The right one is labeled "Kategoria wzrostowa" and is set to "Small". Below these are two buttons: "Dodaj" and "Anuluj/Zakończ".

Rysunek 1.5: AgileWeb – ekran łączenia kategorii wzrostowych i konkurencji

Podobnie funkcjonuje ekran modyfikacji kategorii wzrostowych. Administrator powinien utworzyć wszystkie wymagane kategorie wzrostowe zwracając szczególną uwagę na pola:

minimalna wysokość psa – używane przy sugerowaniu użytkownicy kategorii wzrostowej przy zgłoszeniu oraz przy sprawdzaniu poprawności zgłoszeń;

czasy SCT i MCT – mające udział w liczeniu punktów karnych za czas i ewentualnej dyskwalifikacji;

Po zakończonym przygotowywaniu konkurencji i kategorii wzrostowych administrator powinien je powiązać, tzn. przypisać poszczególne kategorie wzrostowe do konkurencji. Każda konkurencja musi mieć przynajmniej jedną kategorię wzrostową.

1.2.4 Lista startowa (administrator)

Początkowo lista startowa (rys. 1.6) zawiera biegi w kolejności zgłoszeń, a zawodnicy nie mają numerów startowych.

Strona główna / zawody - testowe w test / lista startowa

Lista startowa zawodów testowe w test

Skocz do:

- ◆ [Agility](#)
- ◆ [Jumping](#)
- ◆ [Youngsters](#)
- ◆ [Zerówka](#)

Konkurencja: Agility

Kol.	Nr start.	Przewodnik	Pies	Kat. wzr.	
Small					
1	1	Adminowicz Pierwszy	Azor	Small	usuń / zmień / w dół / w górę
2	6	AAA	BBB	Small	usuń / zmień / w dół / w górę
3	0	test test	Reks	Small	usuń / zmień / w dół / w górę
Medium					
1	2	Jan Kowalski	Burek	Medium	usuń / zmień / w dół / w górę
2	3	1 4	Reks	Medium	usuń / zmień / w dół / w górę
3	4	Adminowicz Pierwszy	Burek	Medium	usuń / zmień / w dół / w górę
Large					

Rysunek 1.6: AgileWeb – ekran listy startowej

System umożliwia automatyczne sortowanie listy w celu zapewnienia odpowiedniego odstępu przed startem tych samych przewodników (reguła *FR-Z-8* z wymagań funkcjonalnych, tabela ??). Przesortowanie listy nada również każdemu zawodnikowi numer startowy (o ile go nie ma).

Jeśli istnieje potrzeba ręcznej poprawy kolejności startów, administrator może dowolnie przesuwając biegi poprzez naciśnięcie “*w górę*” lub “*w dół*”. Jeśli po ostatnim sortowaniu listy zostały dodane zgłoszenia nowych zawodników, to należy nadać im numery startowe – albo poprzez ponowne automatyczne przesortowanie listy albo poprzez odnośnik “*Nadaj numery startowe*”. Uwaga – nie dotyczy to kolejnych zgłoszeń zawodnika już uczestniczącego w zawodach, gdyż system dba o to, aby jedna para przewodnik–pies miała w zawodach ten sam numer startowy w każdym biegu.

Po zakończonych czynnościach modyfikacji listy startowej administrator może sprawdzić poprawność zgłoszeń (szczegółowo omówione w wymaganiu *Sprawdzenie poprawności zgłoszeń*, rozdział ??) lub wyeksportować ją do pliku XML dla aplikacji AgileDesk.

Zarówno ekran listy startowej jak i jego pochodne (tj. ekran listy zgłoszeń, ekran druków dla przebiegów, ekran listy z miejscami na wyniki) posiadają swoje specjalne wersje do druku z alternatywnymi arkuszami styli CSS.

1.2.5 Wyniki zawodów (administrator)

Po zakończonych zawodach administrator musi zaimportować wyniki zawodów z wcześniej wygenerowanego pliku przez aplikację AgileDesk.

Konkurencja: Jumping								
Pies	Przewodnik	Numer startowy	Czas	Punkty karne				
Small, SCT: 20,00s, MCT: 40,00s				błędy	odmowy	za czas	łącznie	
-- brak --								
Medium, SCT: 15,00s, MCT: 30,00s				błędy	odmowy	za czas	łącznie	
1	Azor	Adminowicz Pierwszy	1	8.6	1	0	0	1
	Burek	Jan Kowalski	2					Dis.
Large, SCT: 10,00s, MCT: 20,00s				błędy	odmowy	za czas	łącznie	
-- brak --								

Konkurencja: Zerówka								
Pies	Przewodnik	Numer startowy	Czas	Punkty karne				
Small, SCT: 20,00s, MCT: 40,00s				błędy	odmowy	za czas	łącznie	
-- brak --								

Rysunek 1.7: AgileWeb – ekran wyników zawodów

Operacja sprowadza się tylko do wyboru pliku XML i przesłania go do aplikacji AgileWeb. Wcześniejsze wyniki danych zawodów zostaną usunięte przed importem nowych. W importowanych wynikach mogą znaleźć się biegi zawodników nieposiadających konta w systemie – pojawią się oni na liście wyników zawodów oraz zostanie uaktualniona lista startowa.

Wyniki zawodów (rys. 1.7) dostępne są przez cały czas od ich wprowadzenia. Ekran ten posiada posiadają swoje specjalne wersje do druku z alternatywnymi arkuszami styli CSS tak, aby drukowana postać była czytelna.

Rozdział 2

Aplikacja AgileDesk

2.1 Instalacja

2.1.1 Wymagania

Aplikacja projektowana i pisana była z postawieniem na niezależność od systemu operacyjnego, na którym ma pracować. Przetestowana pod tym względem była na Microsoft Windows XP oraz GNU/Linux, ale jej możliwość uruchomienia jest ograniczona tak naprawdę tylko przez ograniczenia samego Pythona i biblioteki GTK+.

AgileDesk wymaga Pythona w wersji 2.4 lub 2.5. Najprawdopodobniej bez przeszkód uda się ją uruchomić na planowanej wersji przejściowej 2.6 (nieдоступnej w momencie pisania aplikacji). Źródła jak i przygotowane paczki dla różnych systemów operacyjnych i architektur można pobrać ze strony domowej Pythona [14].

Podstawową wymaganą biblioteką jest GTK+ w wersji 2.10 lub nowszej, która odpowiada za graficzny interfejs użytkownika. Na stronie domowej [15] udostępnione są jej źródła. Paczki dla systemów Unix i GNU/Linux należy zainstalować poprzez menedżer oprogramowania danej dystrybucji. Dla systemów Microsoft Windows przygotowane wersje instalacyjne można pobrać ze strony “*Glade/GTK+ for Windows*” [17].

Ostatnimi wymaganymi bibliotekami są interfejsy GTK+ dla języka Python, czyli PyGTK, PyGobject oraz PyCairo. Tak jak w przypadku GTK+ instalacji na systemach Unix i GNU/Linux należy dokonać przez menedżer oprogramowania. Dla Microsoft Windows przygotowane paczki instalacyjne podane są na stronach PyGTK [19].

2.1.2 Instalacja wymaganych komponentów na GNU/Linux

Instalację wszystkich wymaganych komponentów należy przeprowadzić poprzez menedżer oprogramowania danej dystrybucji. Przykładowa instalacja dla systemu Gentoo Linux:

```
# emerge dev-lang/python
# emerge x11-libs/gtk+
# emerge dev-python/pygtk
# emerge dev-python/pygobject
# emerge dev-python/pycairo
```

2.1.3 Instalacja wymaganych komponentów na Microsoft Windows

Po pobraniu archiwów instalacyjnych ze stron wymienionych w rozdziale 2.1.1 należy po kolei zainstalować je w następującej kolejności:

1. Python (np. `python-2.5.1.msi`);
2. GTK+ (np. `gtk-2.10.11-win32-1.exe`);
3. PyGTK (np. `pygtk-2.10.6-1.win32-py2.5.exe`);
4. PyGobject (np. `pygobject-2.12.3-1.win32-py2.5.exe`);
5. PyCairo (np. `pycairo-1.2.6-1.win32-py2.5.exe`).

Przy domyślnej instalacji (bez zmiany miejsca docelowego) pliki poszczególnych bibliotek zostaną umieszczone w katalogach:

1. Python – `C:\Program Files\Python25`;
2. GTK+ – `C:\Program Files\GTK`;
3. PyGTK, PyGobject i PyCairo w podkatalogach Pythona.

Ostatnim krokiem może być dodanie katalogu z interpreterem `python.exe` (np. `C:\Program Files\Python25`) do ścieżki poszukiwań plików PATH dla danego użytkownika. Nie jest to krok konieczny, ale może ułatwić później pracę z programem.

2.1.4 Konfiguracja AgileDesk

Pliki aplikacji AgileDesk należy bezpośrednio skopiować na dysk twardy do wybranego przez siebie katalogu, przykładowo `C:\agiledesk\` (na tym katalogu dalej się będzie opierać niniejsza instrukcja). Aplikacji nie trzeba konfigurować – domyślne ustawienia powinny być odpowiednie. Jeśli jednak użytkownik chciałby lepiej dostosować zachowanie AgileDesk do swoich potrzeb, to znajdzie konfigurację w pliku `C:\agiledesk\config.py` (plik tekstowy w kodowaniu UTF-8). Wśród istotniejszych zmiennych tam się znajdujących wymienić należy:

DEBUG – boolean (True/False); określa czy mają być drukowane na standardowe wyjście dodatkowe komunikaty o wykonywanych czynnościach (przydatne przy debugowaniu);

WINDOW_WIDTH – integer; początkowa szerokość okna GUI;

WINDOW_HEIGHT – integer; początkowa wysokość okna GUI;

POINTS_FOR_CLICK – integer; ilość punktów przyznawana jednorazowo przy naciśnięciu na przycisk popełnienia błędu przez zawodnika;

2.2 Instrukcja użytkownika

2.2.1 Uruchomienie

Na platformie Microsoft Windows AgileDesk należy przejść do katalogu z aplikacją i uruchomić ją poleceniem:

```
cd C:\agiledesk
C:\Program Files\Python\python.exe agiledesk
```

W przypadku pozostałych systemów operacyjnych (tj. GNU/Linux, Unix) wystarczy uruchomić bezpośrednio plik wykonywalny `agiledesk`.

AgileDesk posiada graficzny interfejs do pełnej komunikacji z użytkownikiem, aczkolwiek korzysta również ze standardowego wyjścia do wyświetlania komunikatów o błędach czy wykonywanych czynnościach (przy włączonej opcji `DEBUG`).

2.2.2 Rozpoczęcie pracy

Pierwszą czynnością po uruchomieniu programu musi być wczytanie:

1. listy startowej (CTRL+I lub z menu *Plik – Importuj listę startową*);
2. stanu aplikacji (CTRL+O lub z menu *Plik – Wczytaj stan aplikacji*);

W obydwu przypadkach zostaną wyświetlone okna wyboru pliku. Przez stan aplikacji rozumie się wcześniej utworzony plik poprzez *Zapisz stan aplikacji* (również z menu *Plik*).

2.2.3 Wyświetlenie i modyfikacja listy startowej

Listę startową, po wybraniu z menu *Zawody – Lista startowa*, można modyfikować przez cały czas uruchomienia programu (rys. 2.1). Należy pamiętać o następujących regułach:

Rysunek 2.1: AgileDesk – ekran modyfikacji listy startowej

- lista jest sortowana przy wczytaniu, więc zmiana pola *kolejność* nie spowoduje zmiany kolejności wierszy na samym ekranie dopóki nie zostanie on ponownie wyświetlony;

- rekordy dodawane są początkowo na koniec listy – zostaną umieszczone w prawidłowym miejscu przy ponownym wczytaniu ekranu (tj. wykonaniu sortowania);
- przy modyfikacji nazwiska przewodnika lub nazwy psa w jednym rekordzie, to w gestii użytkownika leży zmiana tych danych w pozostałych startach tego zawodnika;

2.2.4 Wprowadzanie wyników biegów

Ekran dostępny z menu *Zawody – Wprowadź wyniki* (rys. 2.2) wyświetla aktualną listę startową, ale z polami odpowiadającymi poszczególnym błędom, czasowi i dyskwalifikacji w celu wprowadzania wyników poszczególnych biegów.

Agility (dług.: 0.0m)													
Small (SCT: 20.0s, MCT: 40.0s)													
[kolejność]	[nr startowy]	[przewodnik:]	[pies]	[błędy]	[odmowy]	[za czas]	[czas]	[dis.]					
1	6	Jan Kowalski	Maks	0	+	0	+	0.0	licz	0.0	start	dis.	reset
Medium (SCT: 15.0s, MCT: 30.0s)													
[kolejność]	[nr startowy]	[przewodnik:]	[pies]	[błędy]	[odmowy]	[za czas]	[czas]	[dis.]					
1	3	Jan Kowalski	Reks	0	+	0	+	0.0	licz	0.0	start	dis.	reset
2	4	Adminowicz Pierwszy	Burek	0	+	0	+	0.0	licz	0.0	start	dis.	reset
Jumping (dług.: 0.0m)													
Medium (SCT: 15.0s, MCT: 30.0s)													
[kolejność]	[nr startowy]	[przewodnik:]	[pies]	[błędy]	[odmowy]	[za czas]	[czas]	[dis.]					
1	3	Jan Kowalski	Reks	0	+	0	+	0.0	licz	0.0	start	dis.	reset
2	1	Adminowicz Pierwszy	Azor	0	+	0	+	0.0	licz	0.0	start	dis.	reset

Rysunek 2.2: AgileDesk – ekran wprowadzania wyników biegów

Użytkownik może ręcznie wpisać wartość błędu lub nacisnąć przycisk *+* inkrementując pole o standardową wartość (ustaloną w pliku konfiguracyjnym `config.py` w zmiennej `POINTS_FOR_CLICK`, domyślnie 5).

System umożliwi pomiar czasu dla każdego z biegów niezależnie poprzez przyciski *Start/Stop* przy każdym biegu w kolumnie *[czas]*. Jeśli pole *[czas]* jest wypełnione, system może automatycznie obliczyć ilość punktów karnych związanych z przekroczeniem średniego czasu przebiegu (SCT; widoczne przy nazwie kategorii wzrostowej) lub ewentualnie zdyskwalifikować zawodnika po przekroczeniu maksymalnego czasu przebiegu (MCT). W tym celu użytkownik powinien nacisnąć przycisk *licz*.

W przypadku popełnienia pomyłki i chęci wyzerowania wyników danego biegu lub skasowania dyskwalifikacji, należy nacisnąć *reset*.

2.2.5 Wyniki zawodów i eksport

The screenshot shows the AgileDesk application window with the following data:

Agility + Jumping								
Medium (SCT: 15.0s, MCT: 30.0s)								
	[nr startowy]	[przewodnik]	[pies]	[czas]	[błędy]	[odmowy]	[za czas]	[łącznie]
1	3	Jan Kowalski	Reks	0.0	0	0	0.0	0.0
Agility (dług.: 0.0m)								
Small (SCT: 20.0s, MCT: 40.0s)								
	[nr startowy]	[przewodnik]	[pies]	[czas]	[błędy]	[odmowy]	[za czas]	[łącznie]
1	6	Jan Kowalski	Maks	0.0	0	0	0.0	0.0
Medium (SCT: 15.0s, MCT: 30.0s)								
	[nr startowy]	[przewodnik]	[pies]	[czas]	[błędy]	[odmowy]	[za czas]	[łącznie]
1	3	Jan Kowalski	Reks	0.0	0	0	0.0	0.0
2	4	Adminowicz Pierwszy	Burek	0.0	0	0	0.0	0.0
Jumping (dług.: 0.0m)								
Medium (SCT: 15.0s, MCT: 30.0s)								
	[nr startowy]	[przewodnik]	[pies]	[czas]	[błędy]	[odmowy]	[za czas]	[łącznie]

Rysunek 2.3: AgileDesk – ekran wyników zawodów

W dowolnym momencie można wyświetlić aktualne wyniki zawodów (menu *Zawody – Wyniki zawodów*), przy czym nie zawierają one będą biegów z czasem równym 0,0 (rys. 2.3).

W celu przeniesienia wyników do aplikacji AgileWeb należy dokonać ich eksportu do pliku XML. Po wyborze z menu *Plik – Eksportuj wyniki zawodów* (lub skrót klawiszowy CTRL+E) ukaże się okno wyboru pliku do zapisu.

Bibliografia

- [1] PHP: Hypertext Preprocessor. <http://www.php.net>
- [2] PHP License Information. <http://www.php.net/license>
- [3] The Free Software Foundation. <http://www.fsf.org>
- [4] The Free Software Foundation. GPL-Incompatible Free Software Licenses – PHP License, Version 3.01.
<http://www.fsf.org/licensing/licenses>
- [5] CodeIgniter – Open source PHP web application framework.
<http://codeigniter.com>
- [6] CodeIgniter License Agreement.
http://codeigniter.com/user_guide/license.html
- [7] CodeIgniter Installation Instructions.
http://codeigniter.com/user_guide/installation/index.html
- [8] Jesse James Garrett. Ajax: A New Approach to Web Applications.
<http://www.adaptivepath.com/ideas/essays/archives/000385.php>
- [9] xajax PHP Class Library. <http://www.xajaxproject.org>
- [10] xajax – License. http://www.xajaxproject.org/bsd_license.txt
- [11] Python Programming Language. <http://www.python.org>
- [12] Python 2.4.2 license.
<http://www.python.org/download/releases/2.4.2/license/>
- [13] The Free Software Foundation. GPL-Compatible Free Software Licenses – License of Python 2.0.1, 2.1.1, and newer versions.
<http://www.fsf.org/licensing/licenses>
- [14] Download Python Software. <http://www.python.org/download>

- [15] GTK+ – The GIMP Toolkit. <http://www.gtk.org>
- [16] GTK+ FAQ. What is GTK+? [GTK 2.x].
<http://www.gtk.org/faq/#AEN81>
- [17] Glade/GTK+ for Windows.
<http://gladewin32.sourceforge.net/modules/wfdownloads>
- [18] PyGTK: GTK+ for Python. <http://www.pygtk.org>
- [19] PyGTK Downloads. <http://www.pygtk.org/downloads.html>
- [20] MySQL Community Server. <http://www.mysql.com>
- [21] The Apache HTTP Server Project. <http://httpd.apache.org>
- [22] PostgreSQL object-relational DBMS. <http://www.postgresql.org>
- [23] W3C Recommendation. XHTML 1.0 The Extensible HyperText Markup Language (Second Edition). <http://www.w3.org/TR/xhtml1>
- [24] W3C Recommendation. Cascading Style Sheets, level 1.
<http://www.w3.org/TR/REC-CSS1>
- [25] Drew McLellan. Very Dynamic Web Interfaces.
<http://www.xml.com/pub/a/2005/02/09/xml-http-request.html>
- [26] GNU General Public License, version 2
<http://www.gnu.org/licenses/old-licenses/gpl-2.0.html>