

家庭用フィットネス機器を利用した 遠隔トレーニング環境を実現するシステムの設計と実装

橋本 和樹[†] 廣瀬 峻[†] 横山 祥恵[†] 南 政樹[‡] 村井 純[†]

慶應義塾大学環境情報学部[†]

ユビキタス社会におけるインターネットを用いた遠隔トレーニングでは、ユーザから時間、距離などの制約を取り除き、運動を促進することができる。本研究ではトレーナーとユーザが離れている場合でもトレーニングを円滑かつ適切に行なうための機構を設計、そしてその試作を行なった。また、2002 年度に藤沢市で実施された e-ケアタウンプロジェクトにおいて実証実験を行ない、本システムの利用可能性を示した。

The Design and Implementation of Remote Training System using Household Fitness Equipment

Kazuki Hashimoto[†] Shun Hirose[†] Sachie Yokoyama[†] Masaki Minami[†] Jun Murai[†]

[†]Faculty of Environmental Information, Keio University

In the ubiquitous society, remote training using the Internet removes restrictions of time and distance, and promotes training. In this research, a mechanism for training smooth and efficient training even when the trainer and the user are geographically separated, was designed and implemented. In addition, the evaluational experiment was conducted in the healthrise program of e-care town project carried out in Fujisawa City in the 2002 fiscal year, and the availability of this system was evaluated.

1 はじめに

近年、健康増進、体力増強を目的とした家庭用フィットネス機器を利用する人が増えてきている。例えばエアロバイクやダンベルなどがその例である。その理由として、これらの機器は手軽に自由な時に運動をできる点が挙げられる。このような運動機器は特に働き盛りの時間が取れない中高年に利用されているケースが多い。ところが、これらの運動装置は本来アスレチックジムなどで専門家のアドバイスを受け、適切な運動を行なわないと十分な効果が生まれないだけでなく、身体を危険な状態に陥れる可能性もある。

このことから、現代人が家庭で好きな時間を利用して運動でき、アスレチックジムで専門家の指導を受けるのと同じように、専門知識に基づいた適切な運動処方が利用できる仕組みが必要である。

1.1 問題提起

これまでに、PC を介した遠隔トレーニングのための技術が開発されてきた。例えば Combi 社が行っている「クラブ-コンビ.com」[1] は同社の家庭用機器を対象としている。しかしこのようなシステムの多くは、人間が運動装置の制御を行い、そのデータを抽出し送信する作業を行なう必要がある。運動装置を制御するデバイスが、一般的に PC などといったものが前提となっており、特別な作業をする必要があるため、一般の人にはハードルが高い。

2 遠隔トレーニングシステム

本研究で提案する遠隔トレーニングシステムは家庭用フィットネス機器を対象とし、あらかじめ指示された運動メニューに基づき運動装置を遠隔から制御するシステムである。その運動から得られる情報はサーバに保存され、それを元に専門家とユーザがコミュニケーションをとりつつユーザが運動をする。ここで述べるユーザとは運動を行なう者を指す。

本システムは、専門家のアドバイスを受けながら

[†]Keio University Shonan Fujisawa Campus
5322, Endo, Fujisawa, Kanagawa 252, Japan
E-Mail: ia@sfc.wide.ad.jp

運動が行なえるアスレチックジムと同等の実現する。

図 1 に本システムの概念図を示す。

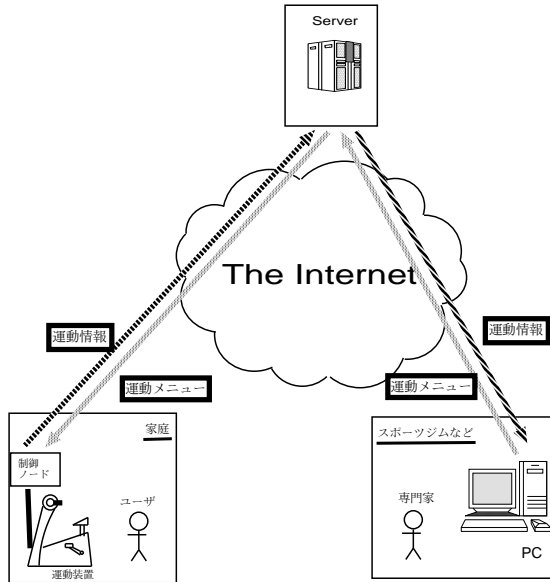


図 1: 概念図

3 前提

3.1 設計要求

本システムではインターネットを通して運動装置を制御できなければならない。そのため運動装置は、外部から制御できる機器でなければならない。

場所によらずにユーザを一意に識別できる仕組みが必要である。どのようなユーザでもその認証を利用できるべきである。このため、一般的なパスワード認証などではなく、電子回路に ID を埋め込むような専用デバイスを使う必要がある。以下このデバイスを認証キーと呼ぶ。

専門家の指示した運動メニューは、ユーザが要求した時に取得できなければならない。そしてその運動メニューで運動装置が制御されると共に、運動をしたデータが専門家の手元に届けられる必要がある。つまり、サーバ上で運動メニュー管理、運動情報蓄積を行なう必要がある。

一方でユーザ側のクライアント（以下、制御ノードと呼ぶ）に必要な機能は、ネットワーク自動設定、認証キー、運動装置の外部制御、双方向通信できる。

ユーザ、専門家の計算機環境の依存性を低減するためにユーザが自らの運動情報の履歴を参照、専門家が各ユーザの運動履歴を閲覧し、運動メニューの編集は容易に行なえなければならない。

3.2 機能要件

設計要求より本システムに必要な機能要件として以下の四点を挙げる。

- 個人認証
運動装置を使用しているユーザを特定できる。
- 運動装置制御
制御ノードがユーザの運動メニューを取得し、それに基づき運動装置を制御できる。
- 運動情報蓄積
取得した運動情報をインターネットを通じて運動情報管理サーバに蓄積できる。
- 運動情報管理
結果をユーザが閲覧でき、専門家が運動メニューの管理を行なえる。
- 運動情報データベース
本システムを使用するユーザ管理、また運動メニュー、運動情報を蓄積できる。

3.3 動作手順

動作手順は以下の通りになる

1. ユーザが認証キーを用い、サーバが認証を行なう。
2. ユーザがトレーニングをする際に各人のメニューがサーバから送信され、そのメニューを元に運動装置が制御され、ユーザが運動を行なう。
3. トレーニング中に運動装置から取得できる運動情報はサーバに送信される。
4. トレーニング後、ユーザは自分のトレーニングのデータ、履歴を WEB から参照する。
5. 専門家はユーザがトレーニングを行なったことを知るとトレーニングデータを閲覧し、ユーザの運動メニューを編集する。

4 設計

3.2 の要件を踏まえて今回家庭用フィットネス機器を用いた遠隔トレーニングシステムの具体的なシステム設計を行なった。

4.1 個人認証

サーバ内にユーザのデータベースを保持し、毎回トレーニング時に制御ノードが認証キーから ID を読みとり、サーバに送信する。その ID を基に、サーバのデータベースにそのユーザがあらかじめ登録されているか調べる事によって認証する。

個人認証の概念図を図 2 に示す。

認証が成功すると、データベースより運動メニューが送られてくる。

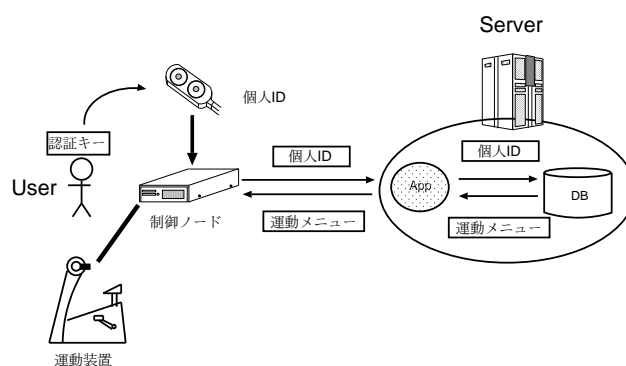


図 2: 個人認証

4.2 運動装置制御

個人認証が成功すると、サーバにあらかじめ登録されたユーザであれば、運動メニューが制御ノードに送信され、その運動メニューに基づいて運動装置を制御する。また、ユーザがトレーニングを始めるとそれを制御ノード側で検知し、運動情報をサーバに送信する。

運動装置制御概念図を図3に示す。

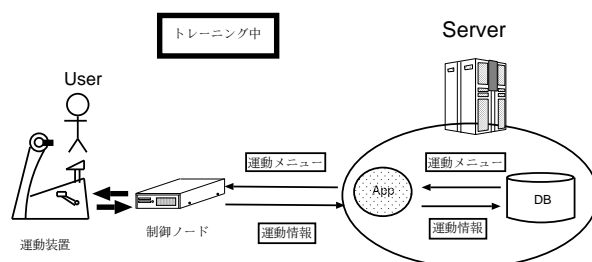


図 3: 運動装置制御

4.3 運動情報管理

運動情報管理で取り扱う情報は下に挙げる四つである。

- 個人情報
- 運動メニュー
- 運動履歴情報
- トレーニング中における運動情報

上記の情報は全てサーバ上で管理される。

その情報は運動履歴として、毎回のトレーニング履歴が詳細に表示される。詳細情報では脈拍や負荷などの情報をグラフ化する。

4.4 運動情報データベース

運動情報データベースに関して、上記の四つの情報がある。それらを効率良く管理するために、リレー

ショナルデータベースを用いた。情報の実体としてはユーザ、運動メニュー、運動履歴、運動情報が存在する。しかし時系列データである運動情報についてはファイルに保存し、データベースのエンティティとしては扱わない。そしてそのファイルの名前、場所を運動履歴テーブルに保持する。以下に示す図4はERモデルを元に作成した概念図である。

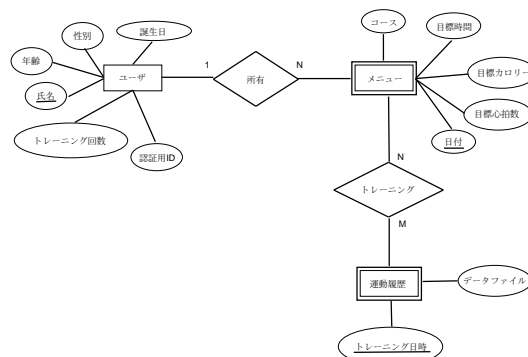


図 4: ER 図-データベース概念図

また、このデータベースにアクセスするモジュールは五つある。以下にそれを示す。

- ユーザ認証モジュール
ユーザテーブルの中からユーザ ID が存在するか検索する。
- 運動メニュー取得モジュール
ユーザ認証で成功した後、ユーザ ID と日付を元にそのユーザの運動メニューをトレーニングメニューテーブルから呼び出す。
- 運動情報保存モジュール
制御ノードから1秒おきに送られてくる運動情報をトレーニングデータテーブルに挿入する。
- 運動履歴表示モジュール
各ユーザが自らの運動履歴を参照する。またトレーニング毎のトレーニング ID を元に詳細なトレーニングデータファイルの場所を検索する。
- 運動メニュー管理モジュール
専門家がユーザの運動メニューを編集する。

上記のモジュールが本システムのデータベースを構成している。これらを効率良く処理するために各テーブル間の関係とプライマリーキーを図5のようにした。下線を引いたパラメータがプライマリーキーである。

図5にデータベースの設計を簡易ER図にして示す。

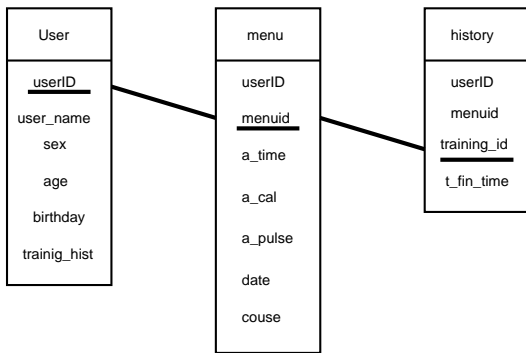


図 5: ER 図-設計

5 実装

本システムの実装について述べる.

5.1 使用機器一覧

まず, 実装したサーバの環境を以下の表 1 に示す.

表 1: サーバ

OS	FreeBSD4.5
Web サーバ	Apache1.3.27[2]
データベース	PostgreSQL7.3.1[3]

次に, 本システムで使用した機器を以下の表 2 に示す.

表 2: 使用機器

運動装置	Combi 社 エアロバイク ai
認証キー	DallasSemiconductorInc. iButton[4]
レセプター	DallasSemiconductorInc. receptor[5]
制御ノード	Internet Node 社 RealSpace6[7]

運動装置には Combi 社のエアロバイク ai, これは PC などから RS232c ケーブル経由で制御コマンドを送信して制御できる自転車エルゴメーターである.

また, 認証キーにはユーザを一意に認識するために iButton と呼ばれる直径 16mm の小さなボタン型の物理デバイスを使用することにした. この iButton は, 小さなステンレスケースの中に電子回路が収められている. レセプターとは 1-Wire[6] と呼ばれる規格で制御ノードと接続するものである. 今回使用した iButton のチップには 64bit の一意な識別子が入っているため, その ID を元にユーザを特定する.

制御ノードにはインターネットノード社の RealSpace6 を使用した. これは IPv6/v4 Dualstack の汎用マイク

ロノードである. OS は Dallas Semiconductor 社の開発している TINI OS[8] が搭載され, JavaVM も実装されている. そのため Java 言語でアプリケーションを開発できる. また, インターフェースには 1-Wire (iButton レセプターと接続), シリアルポート (エアロバイク ai と接続) がついているため本システムに利用可能である.

5.2 データフロー

今回作成したソフトウェアは大きく分けて二つである. 一つはトレーニングを行なう際に動作するソフトウェア, もう一つはトレーニング時間以外に行なうデータの参照や編集である. 前者の流れを図 6 にまとめた.

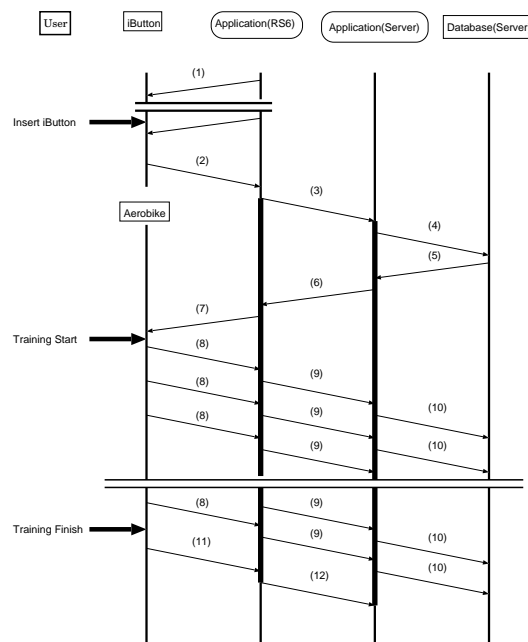


図 6: データフロー図

1. 制御ノードが認証キーが挿入されているかどうかを見張りつつける
2. ユーザが認証キーを挿入するとその ID を読みとる
3. 制御ノードがサーバに接続し, タイムスタンプと認証キーから読み込んだ ID をサーバに送信
4. サーバでは ID を元にデータベースにユーザの認証と運動メニューの取得を試みる
5. 認証が成功するとデータベースから運動メニューを取得する
6. メニューを運動装置制御コマンドに変換して制御ノードに送信する
7. 制御ノードは受け取った制御コマンドを元に運動

装置を制御する

8. トレーニング中は運動情報が運動装置から出力される
9. 取り出した情報を制御ノードからそのままサーバに送信する
10. サーバに送られてきた情報を解析しデータベースのレコードとして保存する
11. トレーニングが終わると運動装置から出力されるステータスが変わる
12. トレーニングが終了したのを制御ノードが検知すると終了コマンドをサーバに送信し、コネクションを切断する

以上がユーザが一回のトレーニングを行なう際の大まかなデータフローである。

5.3 個人認証

1. 制御ノード（クライアント）

起動すると認証キーが挿入されているかを定期的にポーリングし続けるデーモンプログラムが立ち上がるように作った。認証キーより ID を読みとるとサーバに接続し取得した ID とタイムスタンプをサーバに送信する。このプログラムは制御ノード上で Java 言語で実装した。

2. サーバ

サーバは制御ノードからの接続を待つ。制御ノードから ID とタイムスタンプが送られてくるので、それを基にデータベースのユーザテーブルに正しいユーザかどうか、SQL を用いて問い合わせを行なう。

このプログラムは C 言語を用いて実装した。PostgreSQL との係係には libpq[9] というライブラリを使用した。

5.4 運動装置制御

個人認証の成功後、運動メニューを取得し、それを元に運動装置を制御する。

1. メニュー取得

ユーザの運動メニューをデータベースの各ユーザのメニューテーブルから取り出す。日付とユーザ ID を元に運動メニューを取り出す。そして制御ノードに対して送信。

2. 制御

サーバから送信されてきた運動メニューメッセージの中から運動メニューを取り出し、運動装置が理解できるバイトコードでシリアルケーブルに書き出す。また、運動装置から運動中に取得できる

ユーザの生体情報を運動装置から読みとり、サーバへ送信する。これらは Java 言語で実装した。

3. 運動情報受信プログラム

クライアント（制御ノード）から送信されてきた運動情報を各ユーザ ID + 履歴 ID のファル名で運動情報を保存する。データは CSV 形式で保存する。これは C 言語で実装した。

5.5 運動情報管理

ユーザが自らの運動情報を参照し、専門家がユーザのメニューを編集する。

1. 運動情報表示モジュール

本モジュールは HTTP を利用して提供する。保持しているデータベースを参照し、脈拍や負荷などの情報を PHP のグラフ化ライブラリである jgraph[10] を用い実装した。

履歴として最新 10 件の日時、時間、トレーニングメニューを表示した。また、それぞれのトレーニングに対して詳細を表示した。データベースよりデータファイルの場所を取得し、そのファイルよりパラメータ（脈拍と時間、負荷と時間）を読みだしてグラフに表示した。

図 7 は運動履歴表示のユーザインターフェースである。

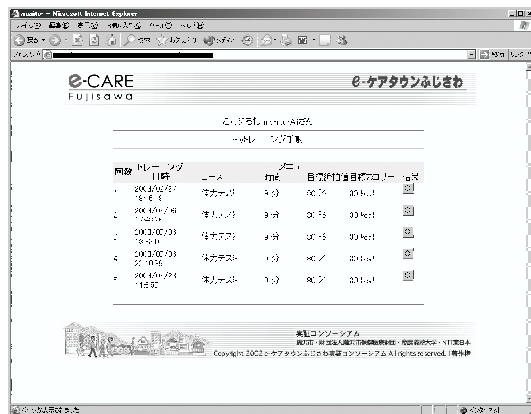


図 7: 履歴情報

また、図 8 に一回のトレーニングの詳細情報表示画面を示す。

2. 運動メニュー管理モジュール

専門家向けに各ユーザの今後の運動メニューを決定し登録するプログラム。PHP を用いて実装した。図 9 はユーザインターフェースのスクリーンショットである。

図 10 は制御ノード、認証デバイス、レセプターを

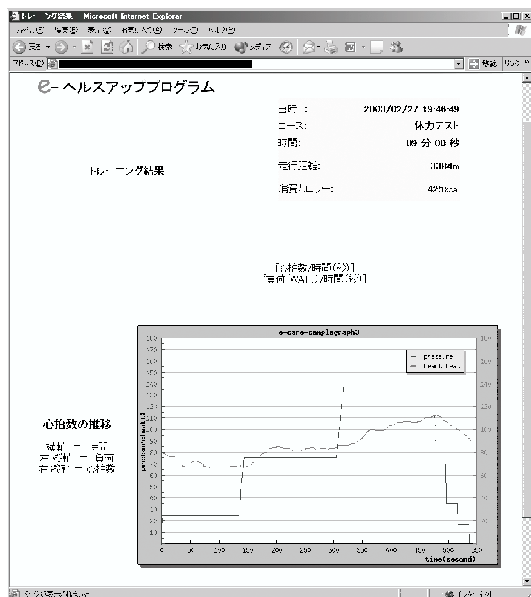


図 8: 詳細情報

積んだ自転車エルゴメーターである。

図 11 は認証キー、レセプターである。

6 実証実験

本研究の遠隔トレーニングシステムを評価するために藤沢市における e-ケアタウンプロジェクト [11] 内で本システムの実証実験を行なった。

6.1 実験内容

藤沢市内の 10 世帯に、光回線を利用した IPv6 常時接続環境を構築し、本システムを設置した。また、慶應義塾大学湘南藤沢キャンパス e-ケアスタジオ内に本システムのサーバを設置した。ユーザの情報を認証用 ID と併せてあらかじめデータベース登録し、各ユーザごとに運動メニューも登録した。そして 2 月 26 日から 3 月 12 日にかけて 2 週間実験を行なった。

6.2 評価項目

遠隔トレーニングシステムの評価項目としては以下の四つを挙げる。今回は本システムがプロトタイプ作成ということもあり、評価では本システムの機能検証に重点をおいた。

各機能の検証項目は以下の四点である。

- 個人認証
認証キーを使い、レセプターに差し込むだけで個人認証が行なわれるか
- 運動装置制御
利用者が個人認証に成功した後に各個人のあらかじめ設定された運動メニューでエアロバイクを制

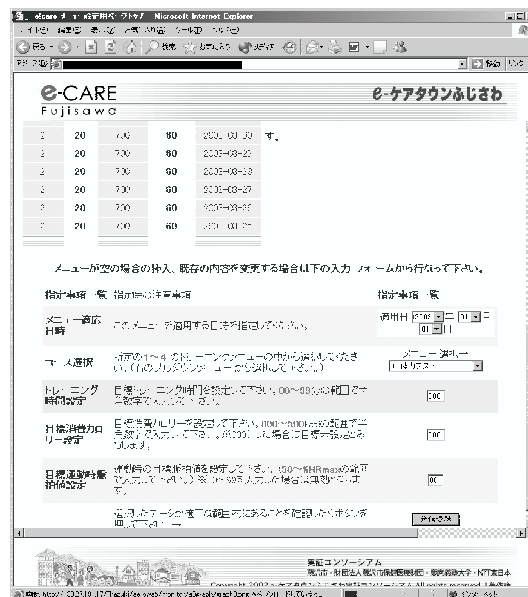


図 9: 運動メニュー管理ユーザインターフェース



図 10: 自転車エルゴメーター (Combi 社エアロバイク ai)

御できたか、またサーバに運動中のデータを送信することができているか

- 運動情報蓄積
運動装置を制御しサーバに送信されてきたデータをサーバ上に保存することができたか
- 運動情報管理
各モニター情報や運動履歴などの情報がデータベースに効率良く蓄積され、ユーザが運動情報を閲覧し、各ユーザごとの運動メニューを編集できているか

6.3 実験結果

1. 個人認証

まず電源を入れ、認証キーをレセプターに挿入す



図 11: 認証キー, レセプター (Dallas Semiconductor 社 iButton)

るとサーバに ID が送信され認証が行なわれるかどうかを検証した。

クライアント側で認証キーが挿入するとサーバにコネクションが張られ ID とタイムスタンプが送信され, その情報を元にデータベースに query が送られ認証が行なわれていることが確認できた。

2. 運動装置制御

認証キーを差し込み, 認証に成功すると運動メニューがサーバから制御ノードに送信されそのメニューを元に運動装置を制御する。

ユーザ宅にて運動装置の液晶画面を見て確認できた。また, トレーニング中のデータを運動装置から読みだしサーバに送信されていることを, サーバ側でパケットキャプチャソフトによりパケットダンプすることにより確認できた。

図 12 はパケットダンプの結果である。

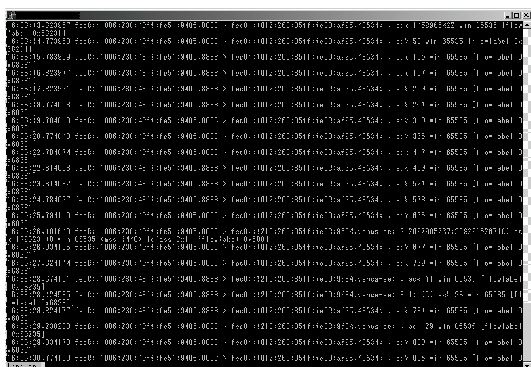


図 12: パケットダンプ画面

3. 運動情報蓄積

ユーザがトレーニングを行なった結果, データベースに履歴がデータベースのレコードに残り, その

履歴情報にマッチするデータがファイルとして保存されていた。

4. 運動情報管理

各ユーザごとに自らの WEB ページを表示させ, 本システムを使用しトレーニングを行なうと, 自動で運動履歴を増やし運動情報を表示することができることを確認した。また, WEB から各ユーザの運動メニューを変更するとデータベース内の各ユーザのメニューが変更されていることが確認できた。

6.4 考察

ユーザは自転車エルゴメーター, その他に電源を入れ, 認証キーをレセプターに差し込むだけであらかじめ専門家に指示された運動メニューでエアロバイクを制御でき, トレーニング結果を自分自身で必要に応じて確認することができた。また, 蓄積したデータを閲覧しユーザに運動メニューを指示することができた。

本研究では, 次世代インターネット技術である IPv6, マイクロノード, 認証デバイス, 自転車エルゴメーターを使用することによって遠隔トレーニングシステムのプロトタイプを作成し, その機能を検証することができたと言える。

7 まとめ

7.1 まとめ

本研究では, 家庭用フィットネス機器を利用した遠隔トレーニング環境実現のためのシステムを設計, 実装した。そして実際のインターネット上で本システムの実験を行なった。

遠隔トレーニングシステムの基本的な機能は実現でき, 実験によってそれが実用に耐えうることが検証できた。

7.2 今後の課題

今回の実験から, 何点か課題が挙げられた。

まずセキュリティである。今回の実験では, ユーザの運動履歴や運動情報は高度な個人情報のため実用化に向けてはセキュリティを考慮しなければならない。

ユーザと専門家のコミュニケーションと言う点ではユーザと専門家の間では運動メニューの編集と運動情報の閲覧でしか行なわれていない。実際にユーザと専門家の間でメールや掲示板などのツールを使用しコミュニケーションを促進しなければならないと考える。

さらに、今回想定したのは専門家一人対ユーザ複数という 1 対 N の環境であるが、実際にサービスとして行なうためには N 対 N のコミュニケーションが発生する。N 対 N のコミュニケーションが行なわれるには統一的な認証方法やコミュニケーションツールが必要となると考えられる。

また、実証実験では 10 人のユーザが本システムを使用したのだが、実際にサービスインする際の想定ユーザ数は遥かに多いため大規模な耐久実験を行なう必要がある。

今回家庭用フィットネス機器の代表ともいえる自転車エルゴメーターを使用したか、本システムは Combi 社のエアロバイク ai しか対応していない。遠隔トレーニングシステムを構築するためには様々な運動装置が使えるなければいけない。現状は運動機器依存になっている。メッセージに対してメタ情報を XML で定義し、様々な機器に対応するプラットフォームを提供すれば、これは解決できる。

さらにトレーニングに関してユーザの目標（健康維持、減量など）でを実現する助けとなることを実証する必要がある。

参考文献

- [1] e-Fitness , <http://www.club-combi.com/efit/intro/index.html> 「Club-Combi.com」
- [2] URL: <http://www.apache.org/> 「The Apache Software Foundation」
- [3] URL: <http://www.postgresql.org/> 「PostgreSQL」
- [4] URL: <http://www.ibutton.com/> 「iButton Homepage」
- [5] DS1402D-DB8: http://www.ibutton.com/ibuttons/blue_dot.html 「iButton receptor」
- [6] URL: <http://www.maxim-ic.com/ibutton/1wirejapanese.htm> 「1-Wire specification」
- [7] URL: <http://www.i-node.co.jp/product/rs6index.html>
- [8] 「The TINI Specification and Developer's Guide」 Don Loomis, 2001
- [9] C Library: the C application programmer's interface to PostgreSQL
- [10] URL: <http://www.aditus.nu/jpgraph/index.php> 「Jpgraph Homepage」
- [11] e care town project, <http://www.e-care-project.jp/>