

GAE 版 MosP インストールマニュアル

Ver1.0(2010/07/05)

Author : GetSet¹ 藤井雄気

GAE 版 MosP Alliance :

ティー・エヌ・エス株式会社²(クラウド ZERO³)

株式会社マインド⁴(MosP⁵)

GetSet(テクニカルサポート)



¹ <http://getset.jp>

² <http://www.tns-g.com/>

³ <http://cloudzero.jp/>

⁴ <http://www.e-mind.co.jp/>

⁵ <http://www.mosp.jp/>

目次

目次.....	2
はじめに.....	3
本マニュアルについて.....	3
プロジェクトについて.....	3
本マニュアルの対象者.....	3
GAE 版 MosP 資材の配置.....	3
前提環境.....	3
GAE プロジェクトの作成.....	4
リリース資材のダウンロードと配置.....	5
build パスの設定.....	6
文字コードの設定.....	7
動作確認.....	7
GAE 環境へのデプロイ.....	10
前提環境.....	10
設定ファイルの記述.....	10
デプロイ.....	10
Tips.....	12
データ削除.....	12
データ投入リンク.....	12
ローカル WEB サーバの停止.....	12

はじめに

本マニュアルについて

このドキュメントはティー・エヌ・エス株式会社が運営するクラウド ZERO サービスのシリーズ第一弾としてリリースされたクラウド版 MosP 勤怠給与システムのインストールマニュアルです。GAE 版 MosP の環境構築手順を提供します。本マニュアルの内容に誤りが見つかりましたら[cloudzero@tns-g.com]までご連絡ください。

プロジェクトについて

MosP は株式会社マインドが提唱した国内初の純国産オープンソース人事給与・勤怠管理システムであり、クラウド版 MosP 勤怠給与システムは MosP の給与計算・勤怠管理パッケージを、Google 社が提供する Google App Engine (GAE) 環境に展開できるように改変しているオープンソースパッケージ (GPL) です。

本マニュアルの対象者

今後、マニュアルや QA を整備していき、どなたでも MosP を GAE 環境に構築できるようにすることを目指しますが、現状では以下のような前提知識が必要になるかと思えます。

- ・ Java についての基本的な知識
- ・ Eclipse (Java IDE) を用いた Java プロジェクトの管理方法
- ・ Eclipse へのプラグインの導入
- ・ GAE アカウントセットアップが可能な程度の英語力
- ・ その他、文字コード、デプロイ、ビルドなどの技術用語と意味

また、本マニュアルでは作業用のローカル端末は Windows 環境であることを想定しています。

GAE 版 MosP 資材の配置

前提環境

以下の環境が作業用端末に存在していることを前提とします。

- ・ Eclipse⁶
- ・ Google App Engien SDK (Eclipse プラグイン)⁷
- ・ その他 JRE、JDK などの Java 実行環境⁸

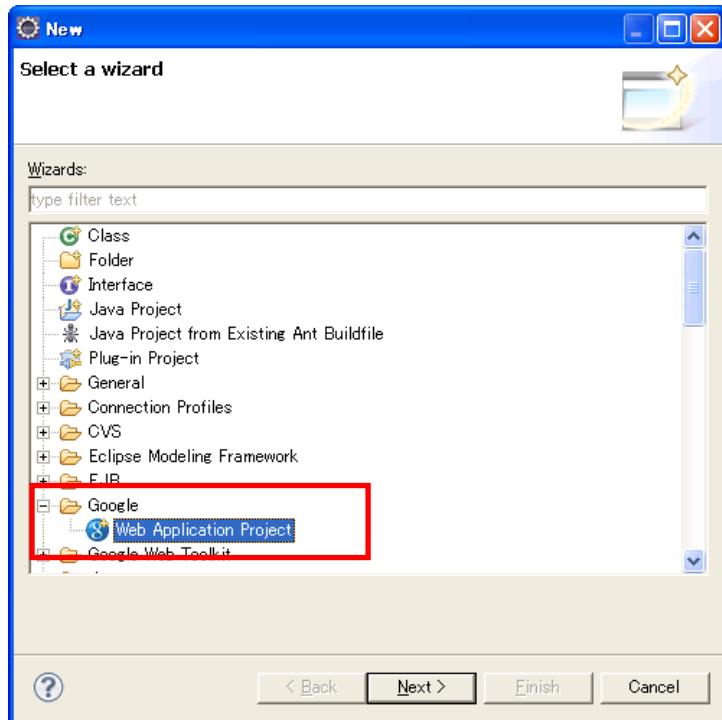
⁶ <http://www.eclipse.org/>

⁷ <http://code.google.com/intl/ja/eclipse/docs/download.html>

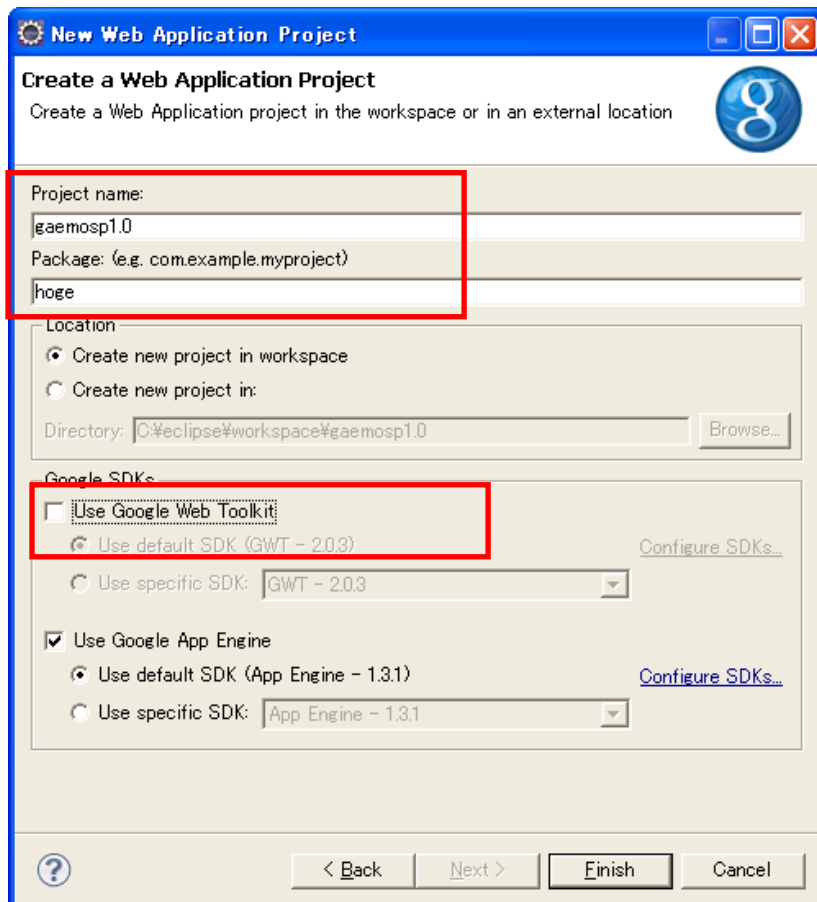
⁸ <http://www.java.com/ja/>

GAE プロジェクトの作成

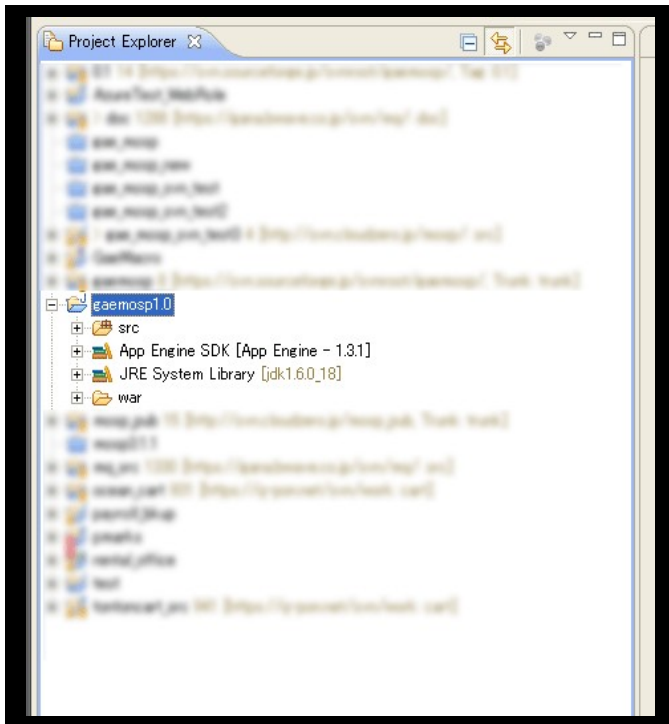
GAE プロジェクトを Eclipse 上に作成します。Eclipse の「File」メニューや「Project Explorer」での右クリックなどから「Google Web Application Project」を選択します。



任意のプロジェクト名、パッケージ名を入力し、Web Toolkit の利用を OFF にして、GAE プロジェクトを作成します。



GAE プロジェクトが作成されます。



リリース資料のダウンロードと配置

SourceForge⁹から最新のリリース資料をダウンロードします。

<http://sourceforge.jp/projects/gaemosp/releases/>

GAE版MosP

概要 ニュース ダウンロード ソースコード Wiki フォーラム メンバーリスト チケット 管理

ダウンロード

オープンソースの人事給与、勤怠管理アプリケーションのMosP(モosp)をGoogle App Engine(GAE)環境で展開できるように改変したGAE版MosPです。

GAE版MosPプロジェクトのリリースは以下からダウンロードできます。

このプロジェクトはオススメ?

リリースファイル一覧

パッケージ/リリース/ファイル	サイズ	日付/ダウンロード数
▼ gaemosp		
▼ 0.1 - [リリースノート / 変更履歴]		2010-07-05 15:42
gaemosp0.1.zip	15.1 MB	

⁹ <http://sourceforge.jp/>

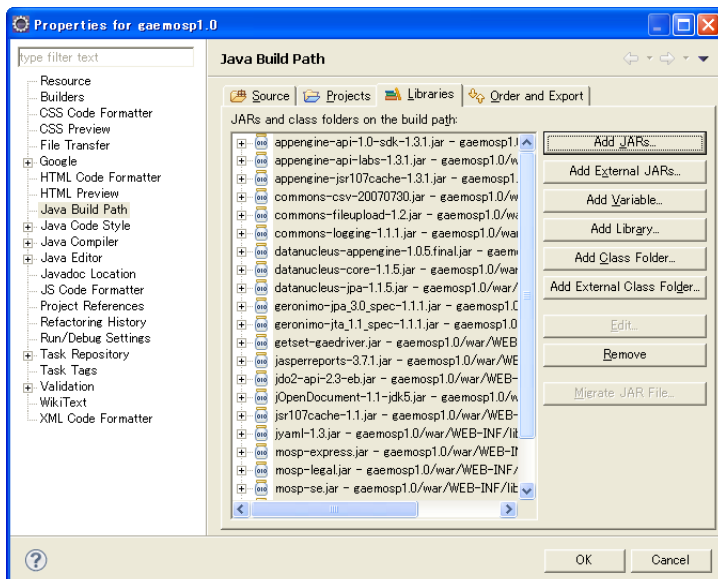
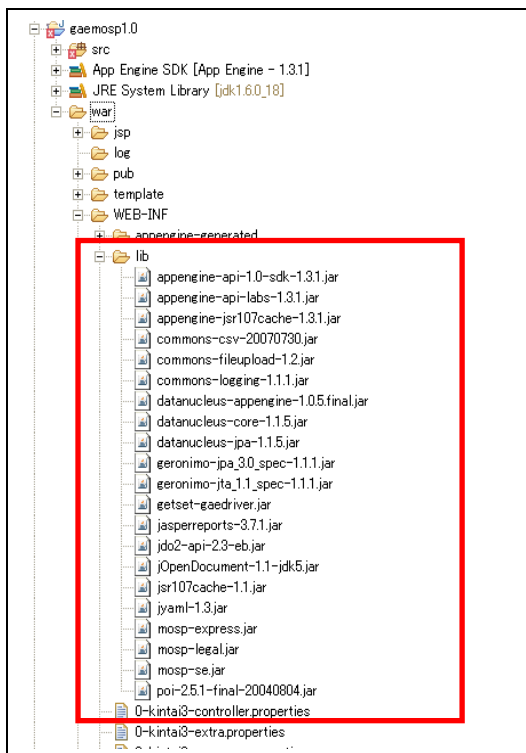
ダウンロードした資料を解凍し、含まれている全ての資料で、Eclipse 上から作成した GAE プロジェクトを上書きします。

ダウンロードした資料で上書き後、Eclipse のプロジェクトを更新し、反映させます。

Eclipse のデフォルトの設定ではコンパイルが始まるはずですが、この段階では必要な資料にパスが通っていないため、エラーが出ます。

build パスの設定

/war/WEB-INF/lib/配下のライブラリ全てに build パスを通します。



文字コードの設定

MosP 資材の Java ソースコードは Windows-31J で記述されており、クラウド ZERO 関係の資材は UTF8 で記述されているため、エンコーディングの設定がやや複雑になります。/src/配下は Windows-31J エンコードをデフォルトに設定し、「jp.mosp」以外の資材は UTF8 でエンコードするように設定して下さい(/src/META-INF/配下も UTF8 です)。

GAE 版 MosP の文字コードは各種のエンコーディングが混在しています。当面 Java 資材がコンパイルできれば問題ないため、踏み込んだ設定は行いませんが、ソースコードの解析や改変の際には適当な設定が必要になってきます。文字コードの設定を脚注にて記載しておきます。¹⁰

文字コードを設定し、エラーが出ていなければローカル上でのプロジェクトの配置は完了です。¹¹

動作確認

GAE 環境にアップロードする前に、アプリケーションの検証をローカルの作業用端末で行います。

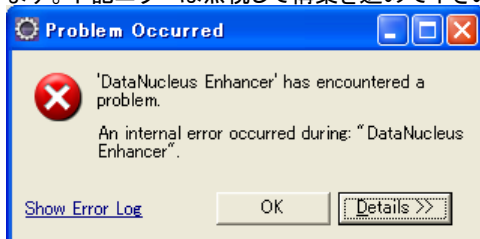
GAE プロジェクトを右クリック、メニューから「Run As > Web Application」を選択、実行し、ローカルでの WEB サーバを立ち上げ、「http://localhost:8888/」にアクセスします。

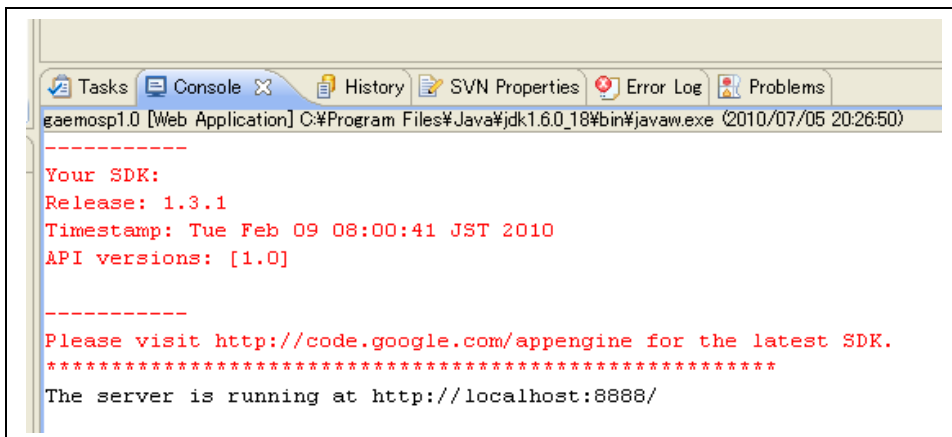
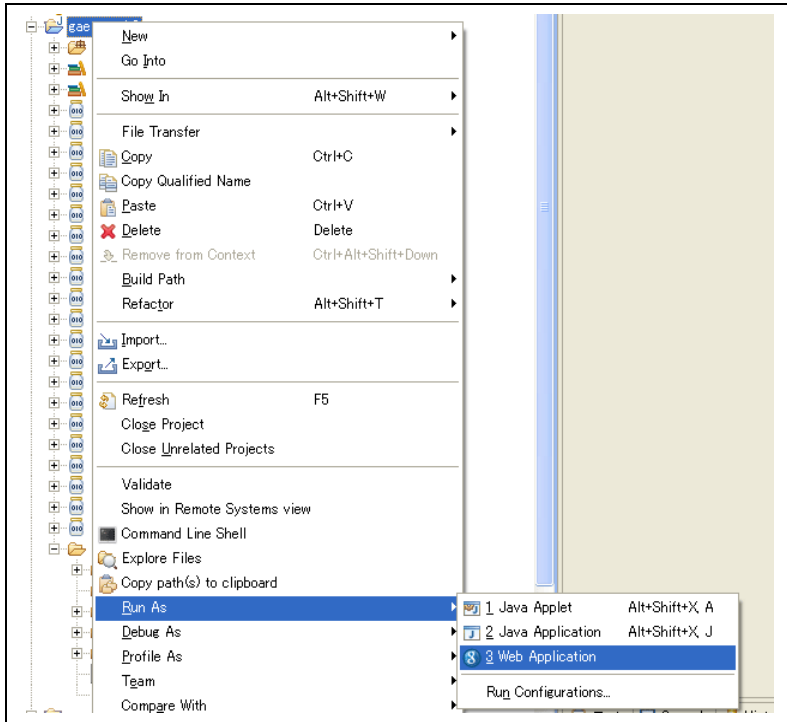
¹⁰ MosP のソースコードは基本的に Windows-31J で記述されています。MosP 公式ドキュメント内では Windows-31J と SJIS、CP932 を区別せずに全て「SJIS」と記載されていますが、Windows-31J・SJIS 間で Java の内部文字コードである Unicode へのコードポイントが異なる、互換性のない文字(例えば「~」など)が Java ソースコード中にハードコーディングされていたりするため、エンコーディングは SJIS ではなく Windows-31J の方が無難です。

css や javascript など SJIS ではなく Windows-31J であるようですが、MosP 公式ドキュメントでは WEB サーバや APP サーバなどのレスポンスヘッダを SJIS に設定するような記述が見られます。css や javascript に関しては、ASCII 範囲しか扱わないため、当面問題はないようです。上述のように諸々の問題が出たり、複雑性が増すため、文字コードは全て UTF8 で統一して頂きたい旨、MosP 開発元である株式会社マインド様には伝えており、次期メジャーバージョンアップ以降での UTF8 対応を検討して頂いております。

今回のクラウド版 MosP に関するソースコードは UTF8 で統一してエンコーディングしておりますが、結果、Windows-31J、SJIS、UTF8 が混在するリリース資材になっています。

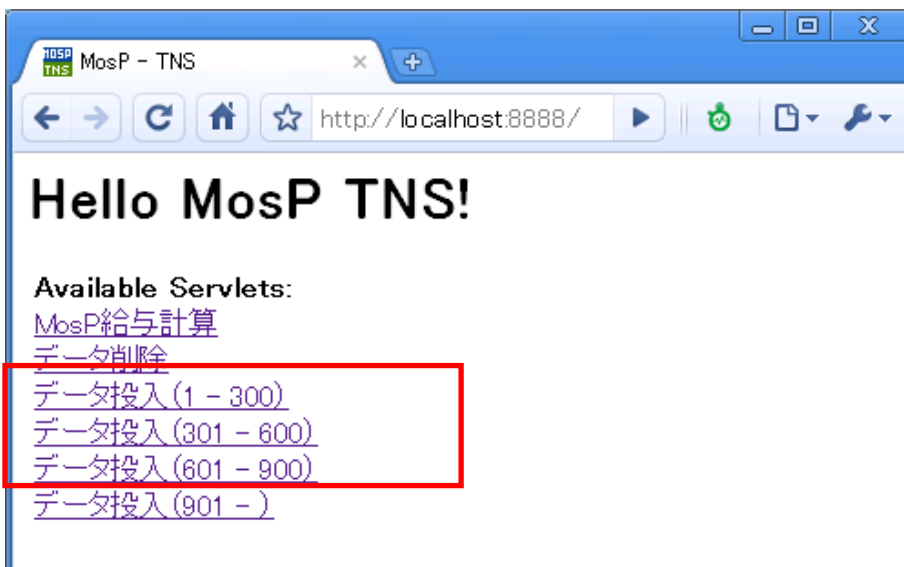
¹¹ 下図のようなエラーが発生する可能性があります。原因は特定できておりませんが、ローカルの Eclipse でのジョブの実行に際して発生しているようで、今のところ GAE 版 MosP 本体の動作への影響は確認されておりません。確認はありませんが、GAE の SDK での不具合だと思われる。下記エラーは無視して構築を進めて下さい。



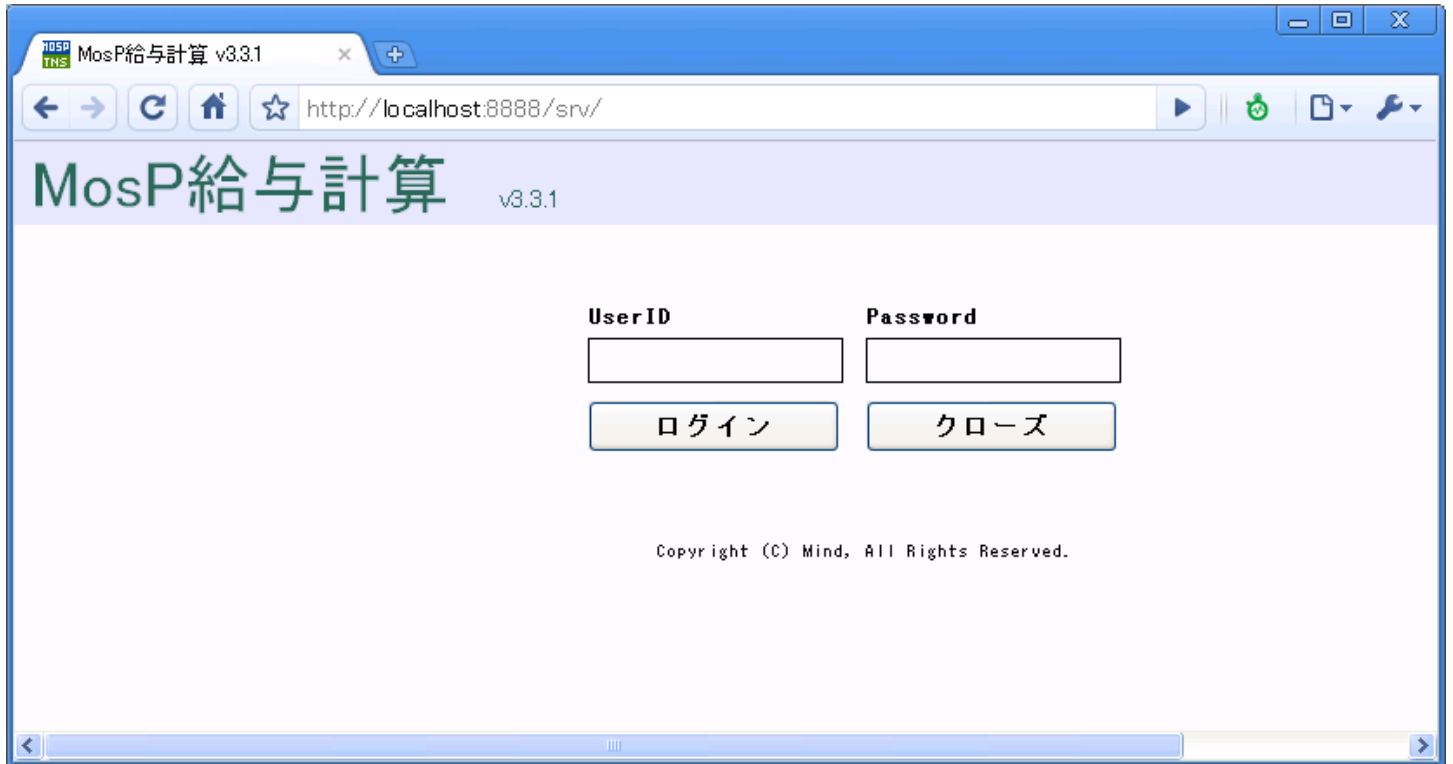


下記画面が表示されれば、WEB サーバが起動しています。

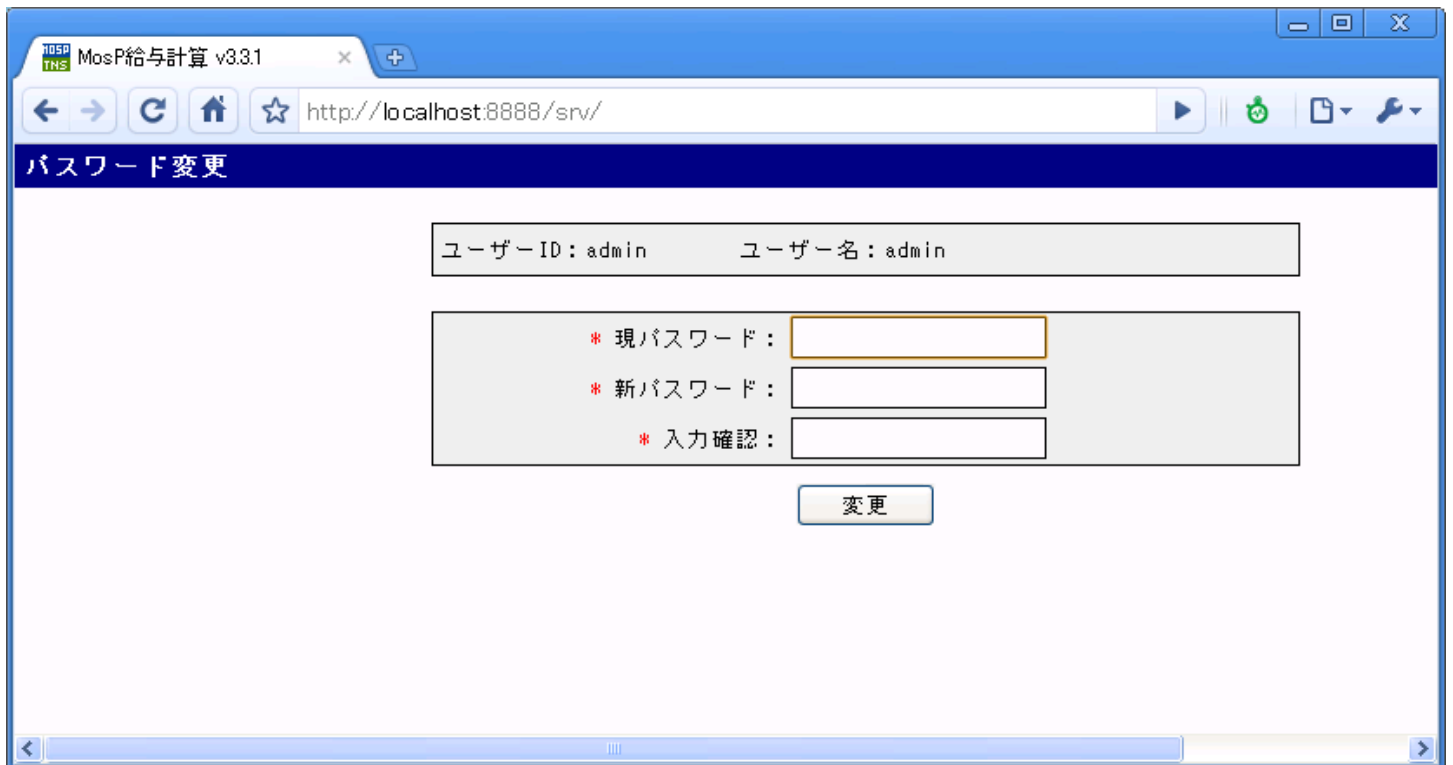
データ投入リンク 3 箇所 (1~900) をクリックし、初期データを構築の後、「<http://localhost:8888/pub/>」にアクセスして下さい。



下記画面が表示されれば GAE 版 MosP が動作しており、利用可能な状態になっています。



デフォルトのアカウント「UserID:admin」「Password:admin」を入力し、パスワード変更画面に進み、クラウド版 MosP 勤怠給与システムの動作確認を行って下さい。



GAE 環境へのデプロイ

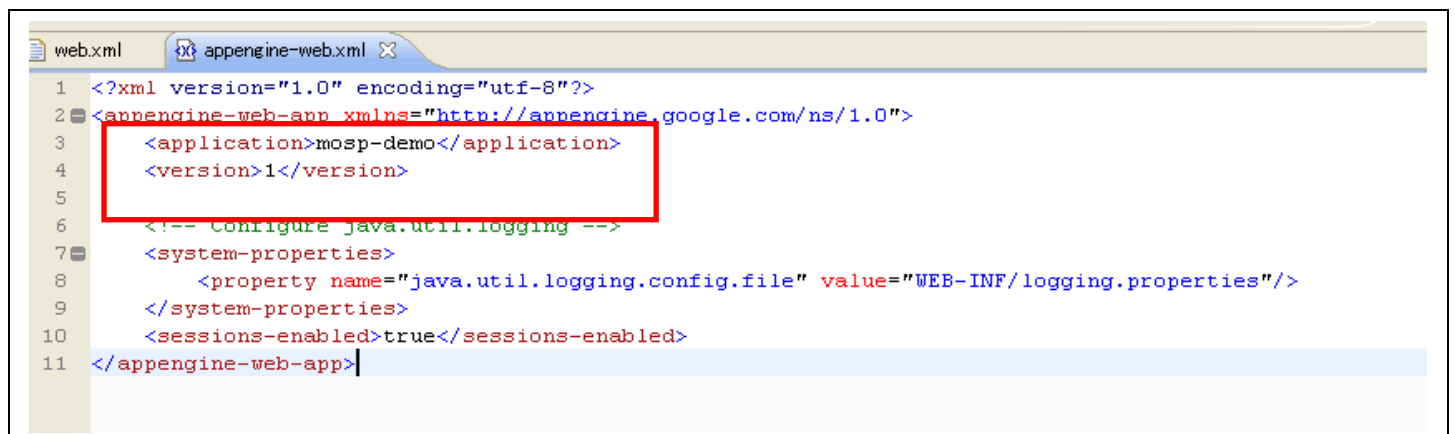
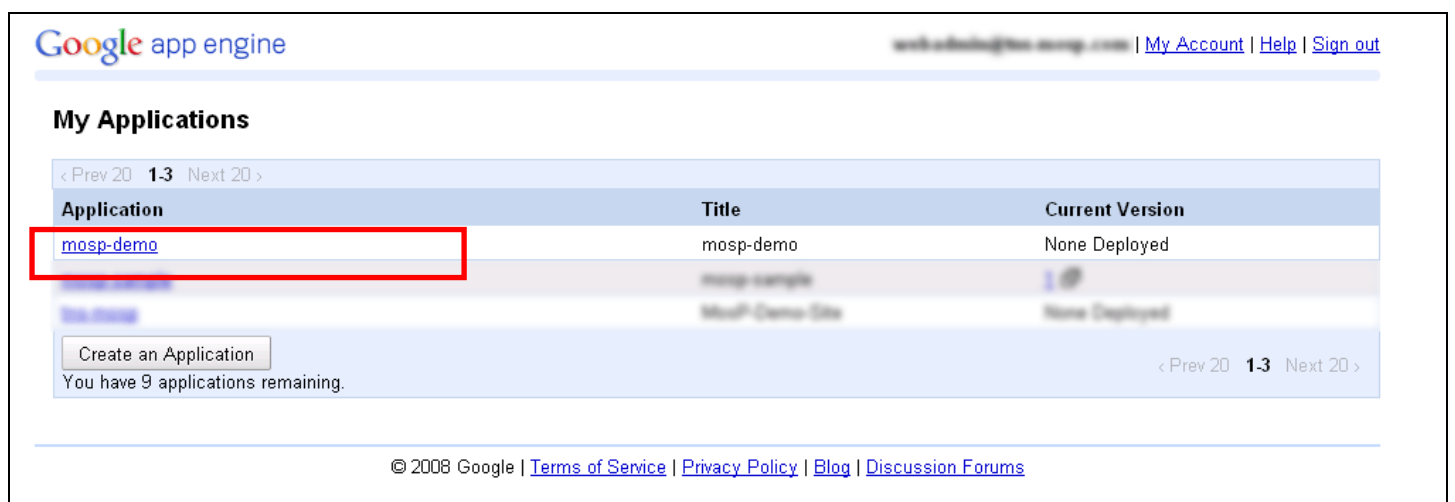
前提環境

Google App Engine のアカウントが存在し、ローカル環境でクラウド版 MosP が動作していることを前提とします。

設定ファイルの記述

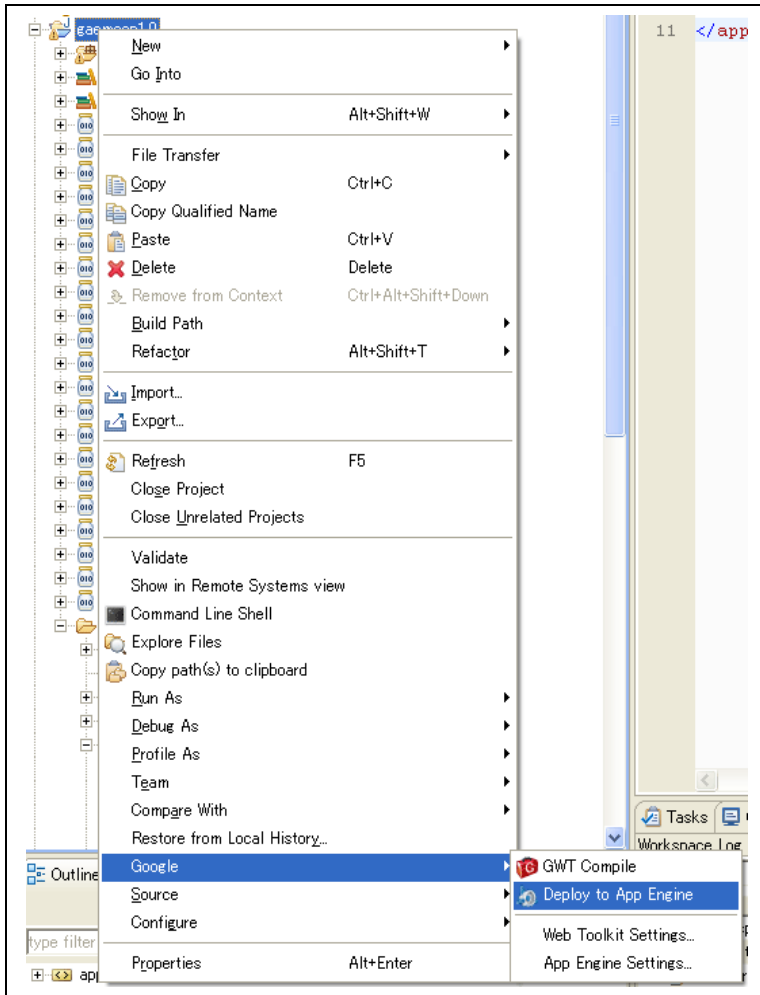
/war/WEB-INF/appengine-web.xml に GAE アプリケーションの記述を追加します。

取得済みのアプリケーション ID を<application></application>に、任意のバージョンを<version></version>に記述します。

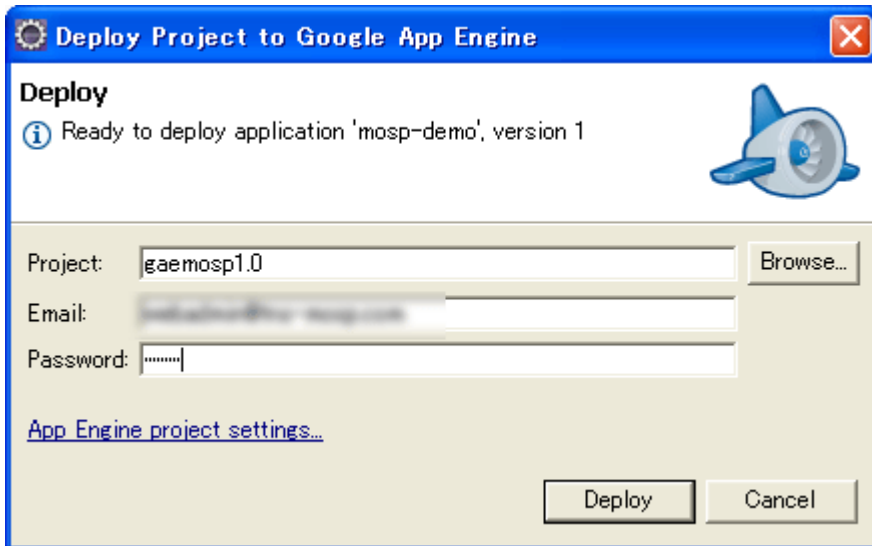


デプロイ

GAE プロジェクトを右クリックし、「Google > Deploy to App Engine」メニューから GAE プロジェクトのデプロイを実行します。

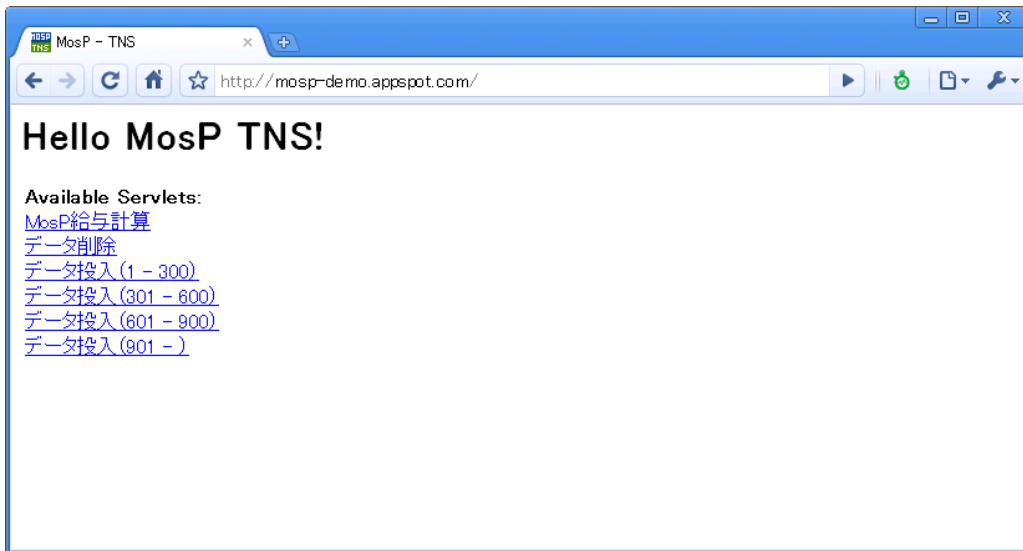


Google アカウント情報を入力した後、Deploy ボタンを押下し、GAE サーバへのアップロードとデプロイを実行します。



デプロイが終了すればアプリケーションが利用可能な状態になっています。

<http://<Application-ID>.appspot.com/>にアクセスし、GAE 版 MosP が動作していることをご確認下さい。



Tips

データ削除

GAE の制約により、データの削除は 1 トランザクションで 500 件までしか行えません。500 件を超えるデータが登録されると、トップページのデータ削除リンクはエラーを返すようになります。データ削除が出来なくなれば GAE コンパネの Datastore Viewer から手動で削除してください。

データ投入リンク

データ投入済みでデータ投入リンクが必要ない場合は /war/index.html からリンクを削除したり、サーブレットを削除したりするなどして対応して下さい。

/war/index.html

⇒ 削除

/war/WEB-INF/web.xml

⇒ `<welcome-file-list>` から `<welcome-file>index.html</welcome-file>` を削除

⇒ `<servlet-name>Init</servlet-name>` のサーブレット定義とサーブレットマッピングを削除

ローカル WEB サーバの停止

ローカルの WEB サーバは Eclipse のコンソールから停止してください。(Terminate ボタン)

WEB サーバを起動したまま次の WEB サーバを起動しようとする、「Address already in use」のエラーが出ます。エラーが出た場合、Eclipse コンソールからの WEB サーバの停止ができなくなりますが、Windows タスクマネージャから javaw.exe プロセスを終了させることで WEB サーバの停止が可能です。