

# チーム開発演習

オブジェクト指向プログラミング及び演習2

第11回 課題

# 今回やること

- Webインターフェイスを持った画像処理システムの構築
- 以下、構築を行うシステムのざっくりとした要件
  1. ユーザーがWebページより画像をアップロードできる
  2. 別プログラムにてアップロードディレクトリを監視し、アップロードされた画像を決められた方法で処理し保存
  3. ユーザーはアップロードされた画像と処理済みの画像をいつでもWebページ経由で閲覧できる
- 上記1～3は、一連のアクションではなくそれぞれが独立していること

# Webプログラムに関して

- Web側は、Flaskを用いて作成します。
- Web側でやることは、あくまでもファイルのアップロードを受け付ける窓口と、処理された画像の一覧を表示することのみです。
- Web側は、以下の2つの作業に分割して作業を割り振ります。
  1. トップページ及びファイルのアップロード部分の作成
  2. アップロード画像や処理済み画像の一覧ページの作成

# 画像処理の内容に関して

- 画像処理には、PythonからOpenCVを使用します。
  - 以下の処理から、人数に好きなものを選んで実装してください
1. 顔検出してモザイク
  2. 顔検出して枠で囲う
  3. Cannyフィルタによる輪郭抽出
  4. 画像のグレースケール化(できれば2値化も)
- 肝心のディレクトリの監視は、ファイルの新規作成、変更を監視してください。それぞれのイベントで、上記の中から選んだ画像処理をすべて行います。  
監視にはWatchdogという外部ライブラリが便利に活用できると思います。

# 演習の進行に関して

- ざっくりと流れの説明をしておきます。
  1. チーム内で話し合い、各メンバーの担当部分を決める
  2. Web側(アップロード、閲覧)をどうやって作るか、画像処理プログラム(ディレクトリ監視)をどうやって作るかを相談する
  3. 作成
  4. 各メンバーは、自分の作成箇所を必ず動作確認した上で PullRequestを作成する
  5. メインブランチで結合テスト

# チームリーダーのお仕事

- 今回のシステム開発のプロジェクト進行を管理してください
  - ※コード書くのは最小限に！
- やること
  - 「今回やること」の1, 2, 3の要件を、チームメンバーに割り振る。
    - だれがどこを担当するかは必ず記録しておいてください。
  - 今回のプロジェクトを保存するリポジトリを作成する。
    - README.mdに、システムの動作確認方法や、使用するライブラリのバージョン等をまとめておきましょう。
  - チームの各プログラムの進行状況を定期的を確認し、適宜相談に乗ること。

# その他メンバーのお仕事

- チームリーダーと相談の上決めた担当部分のプログラムを、新しく作業ブランチを切って頻繁にPushする。
- ある程度形になったタイミングで、動作確認をした上でPull requestを作成してメンバーにレビューをお願いすること。

# 提出物に関して

- 以下の内容をMarkdownで書いてMoodleに提出してください。  
(ファイル名は、K00000\_11.md K00000は自身の学籍番号)
  - GitHubユーザー名
  - 担当箇所
    - 詳しく書くこと
  - チーム作業のログ
    - リポジトリの作業ディレクトリで以下のコマンドを実行、スペースキーを何度か押して、ログすべて表示した上でコピペする
    - `$ git log --graph`
  - 感想
    - 次に活かせる反省等あればここに書いておくと良い