DEVELOPING SOFTWARE USING PYTHON AND DJANGO TO SOLVE REAL LIFE PROBLEMS

ANNA MAKARUDZE

PYCON NAMIBIA 2016 WORKSHOP

29 JANUARY 2016

WORKSHOP STRUCTURE

- Background
- Objectives
- Problem Identification
- Proposed Solution
- System Design
- System Implementation
- Conclusion

BACKGROUND

MANY PROGRAMMERS NEW TO PYTHON ASK THEMSELVES THESE QUESTIONS, SOON AFTER LEARNING THE LANGUAGE: “NOW THAT I HAVE LEARNT PROGRAMMING WITH PYTHON, HOW CAN I USE IT? WHAT SOFTWARE PROGRAMS CAN I DEVELOP WITH IT?”

OBJECTIVES OF WORKSHOP

BY THE END OF THIS WORKSHOP, PARTICIPANTS SHOULD BE ABLE TO:

- IDENTIFY PROBLEMS AROUND THEM THAT CAN BE COMPUTERISED
- PROPOSE A SYSTEM/SOLUTION TO ADDRESS THE PROBLEM
- DESIGN POSSIBLE SOLUTIONS TO ADDRESS THE PROBLEM
- USE PYTHON AND DJANGO TO DEVELOP SOFTWARE/SOLUTION

PROBLEM IDENTIFICATION

- IDENTIFY THE PROBLEM BY LOOKING AROUND YOU
- HOW IS THIS DONE?
- IS THIS THE BEST (MOST EFFICIENT) WAY TO DO THIS?
- WHAT IF WE CHANGED HERE OR THERE?
- SOFTWARE IDEAS/SYSTEMS DO NOT START FROM FICTION MOVIES, THEY EMERGE FROM REAL LIFE, DAY-TO-DAY PROBLEMS!

ACTIVITY 1 – PROBLEM DEFINITION

IDENTIFY A PROBLEM THAT AFFECTS YOU IN YOUR DAY-TO-DAY LIFE. IT MAY BE AS A RESULT OF A LACK OF AN AUTOMATED SYSTEM, INEFFICIENT SYSTEM OR POORLY DEFINED PROCESSES. THE PROBLEM SHOULD BE SOLVABLE OR THERE SHOULD BE ROOM FOR IMPROVEMENT BY AUTOMATION.

PROPOSED SOLUTION

- ANALYSE THE PROBLEM
- CAN THE PROBLEM BE SOLVED?
- HOW CAN WE SOLVE THIS PROBLEM OR IMPROVE THE SITUATION?
- WHAT ARE THE ALTERNATIVES AVAILABLE?
- WHAT RESOURCES DO WE HAVE TO SOLVE THIS PROBLEM?

ACTIVITY 2 – PROPOSED SOLUTION

PROPOSE A SOLUTION/SYSTEM THAT CAN BE USED TO ADDRESS THE PROBLEM THAT YOU JUST IDENTIFIED IN ACTIVITY 1. YOU CAN PROPOSE TWO OR THREE DIFFERENT SOLUTIONS TO THE PROBLEM AND CHOOSE THE ONE YOU THINK SOLVES THE PROBLEM BEST OR IS FEASIBLE AT THIS STAGE.

SYSTEM DESIGN

- COVERS THE FUNCTIONALITY THE SOLUTION WILL HAVE
- WILL IT BE DATABASE-DRIVEN?
- WHICH DATABASE SYSTEM WILL IT USE?
- HOW WILL THE DATABASE TABLES BE STRUCTURED?
- HOW WILL USER INPUT BE CAPTURED?
- MENUS, USER INTERFACE, SECURITY, USER FEEDBACK, PROCESSES/PROCEDURES, ETC.

ACTIVITY 3 – SYSTEM DESIGN

DESIGN THE SYSTEM YOU JUST PROPOSED IN ACTIVITY 2. DESIGN THE DATABASE (IF REQUIRED), WEB PAGES REQUIRED, FORMS REQUIRED, ETC. FOR THE SYSTEM TO BE ABLE TO MEET USER REQUIREMENTS. LIST AND DEFINE ALL THE TASKS TO BE PERFORMED BY YOUR APPLICATION. FOR EACH TASK, WRITE AN ALGORITHM/PSEUDOCODE TO SHOW STEPS TO BE FOLLOWED TO ACCOMPLISH THE TASK.

SYSTEM IMPLEMENTATION

- DEVELOP THE SYSTEM IN THE CHOSEN DEVELOPMENT ENVIRONMENT
- MAY RESULT IN CHANGES TO THE INITIAL DESIGN WITH RESPECT TO DATABASE, MENUS, USER INTERFACE, USER FEEDBACK, PROCEDURES ETC.
- MAKE ADJUSTMENTS AS NECESSARY, ENSURING THE REQUIRED OUTCOME IS ACHIEVED.

ACTIVITY 4 – SYSTEM IMPLEMENTATION

SET UP YOUR APPLICATION IN THE IDE OF YOUR CHOICE. INSTALL DJANGO AND OTHER PACKAGES REQUIRED BY YOUR APPLICATION. DEVELOP AND TEST THE SYSTEM. (THE WORKSHOP ASSUMES YOU HAVE PYTHON INSTALLED ALREADY!)

SETTING UP OUR DJANGO APPLICATION

CREATE AND ACTIVATE A VIRTUALENV

- OPEN COMMAND PROMPT
- TYPE IN THE FOLLOWING COMMANDS

```
virtualenv env
source env/bin/activate
env/Scripts/activate (for Windows users)
```

SETTING UP OUR DJANGO APPLICATION

- INSTALL DJANGO IN THE VIRTUALENV

  pip install django

- CREATE A NEW DIRECTORY FOR THE PROJECT

  mkdir myproject

- CD INTO THE NEW DIRECTORY

  cd myproject

SETTING UP OUR DJANGO APPLICATION

- CREATE A NEW DJANGO PROJECT IN THE FOLDER

  django-admin startproject myproject

- RESULT :

  myproject/

    manage.py

    myproject/

      __init__.py

      urls.py

      settings.py

      wsgi.py

SETTING UP OUR DJANGO APPLICATION

- RUN

  python manage.py migrate

- RUN

  python manage.py runserver

- IN YOUR BROWSER, TYPE http://127.0.0.1:8000

SETTING UP OUR DJANGO APPLICATION

- CREATE YOUR APP

  python manage.py startapp app

- RESULT:

  app/

    migrations/

    __init__.py

    admin.py

    apps.py

    models.py

    tests.py

    views.py

SETTING UP OUR DJANGO APPLICATION

- EDIT YOUR __init__.py
- ADD THE FOLLOWING:

```
"""
Package for app.
"""

default_app_config = 'app.apps.MyProjectConfig'
```

SETTING UP OUR DJANGO APPLICATION

- ADD YOUR APP TO INSTALLED APPS IN YOUR settings.py

```
# Application definition


INSTALLED_APPS = (
    'django.contrib.admin',
    ...
    'app',
)
```

DEVELOPING OUR DJANGO APPLICATION

- FILES TO BE WORKED ON:

  models.py – create our models here

  admin.py – register our models here

  views.py – write our views here

  urls.py – register the urls for our views here

- NEW FOLDERS TO BE CREATED

  templates – create our HTML files/templates here

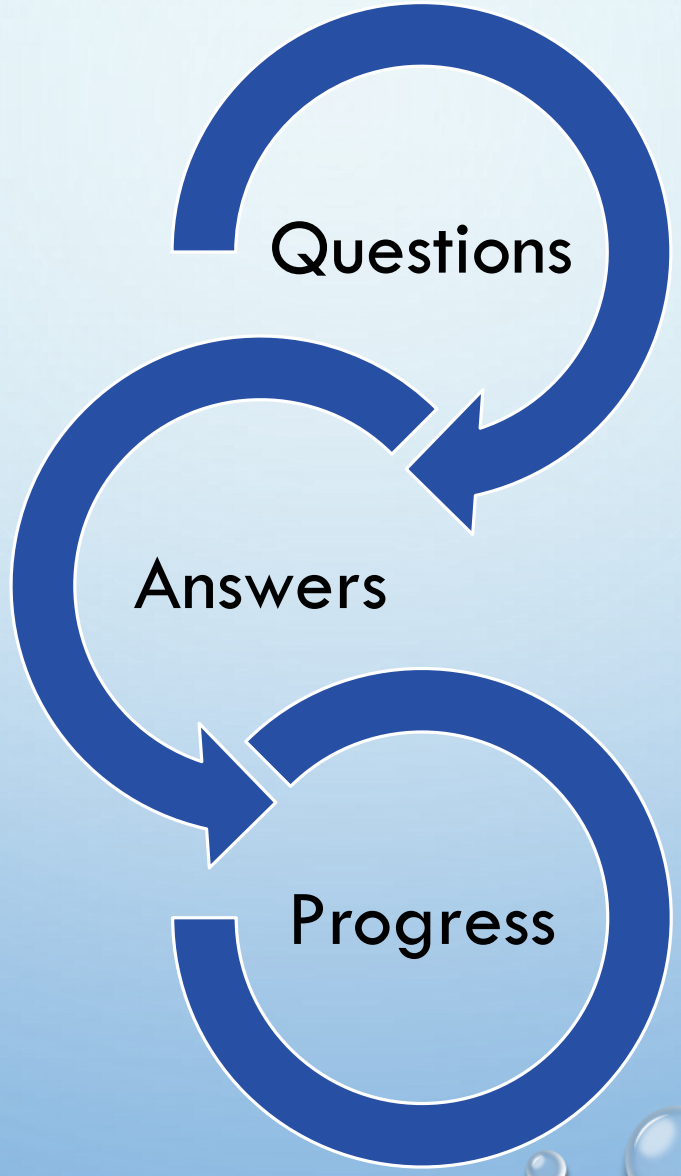
  static – create our CSS files here for styling our pages

DEVELOPING OUR DJANGO APPLICATION

- HTML TEMPLATES REQUIRED:
  - BASE/LAYOUT
  - INDEX
  - AND ANY OTHER PAGES REQUIRED BY YOUR APP

CONCLUSION

- PYTHON/DJANGO CAN BE USED TO DEVELOP SOFTWARE TO SOLVE REAL LIFE PROBLEMS JUST LIKE ANY OTHER PROGRAMMING LANGUAGE.
- THE TASK IS UP TO YOU, THE PROGRAMMER TO DECIDE HOW YOU CAN USE THEM!


Questions
Answers
Progress

END OF PRESENTATION

THANK YOU!
TATENDA!
MUITO OBRIGADO!