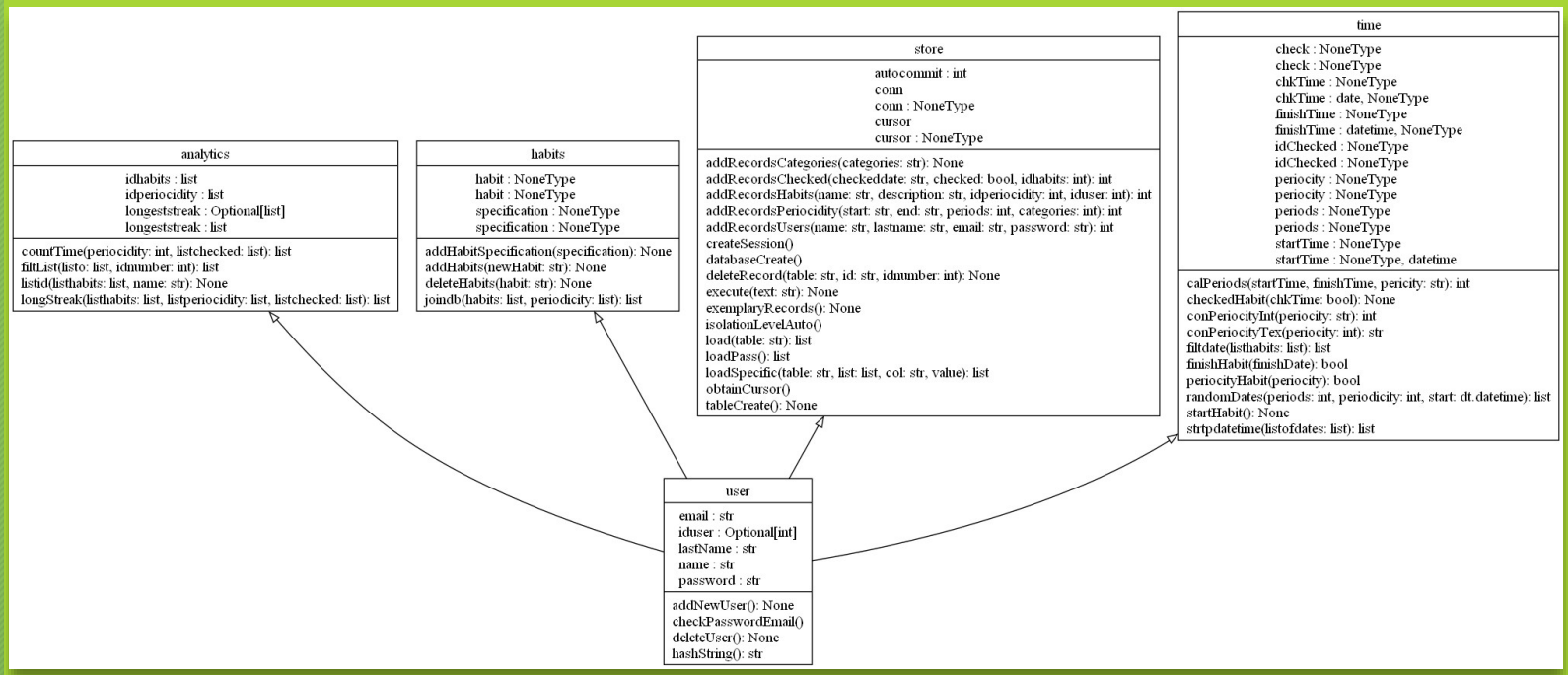
Object Oriented and Functional Programming with Python(DLBDSOOFPP01)

1

Presented by: Jorge E. Caballero
BSc. MSc. Geology – Student Computer Science
Matricule Number: 92014609

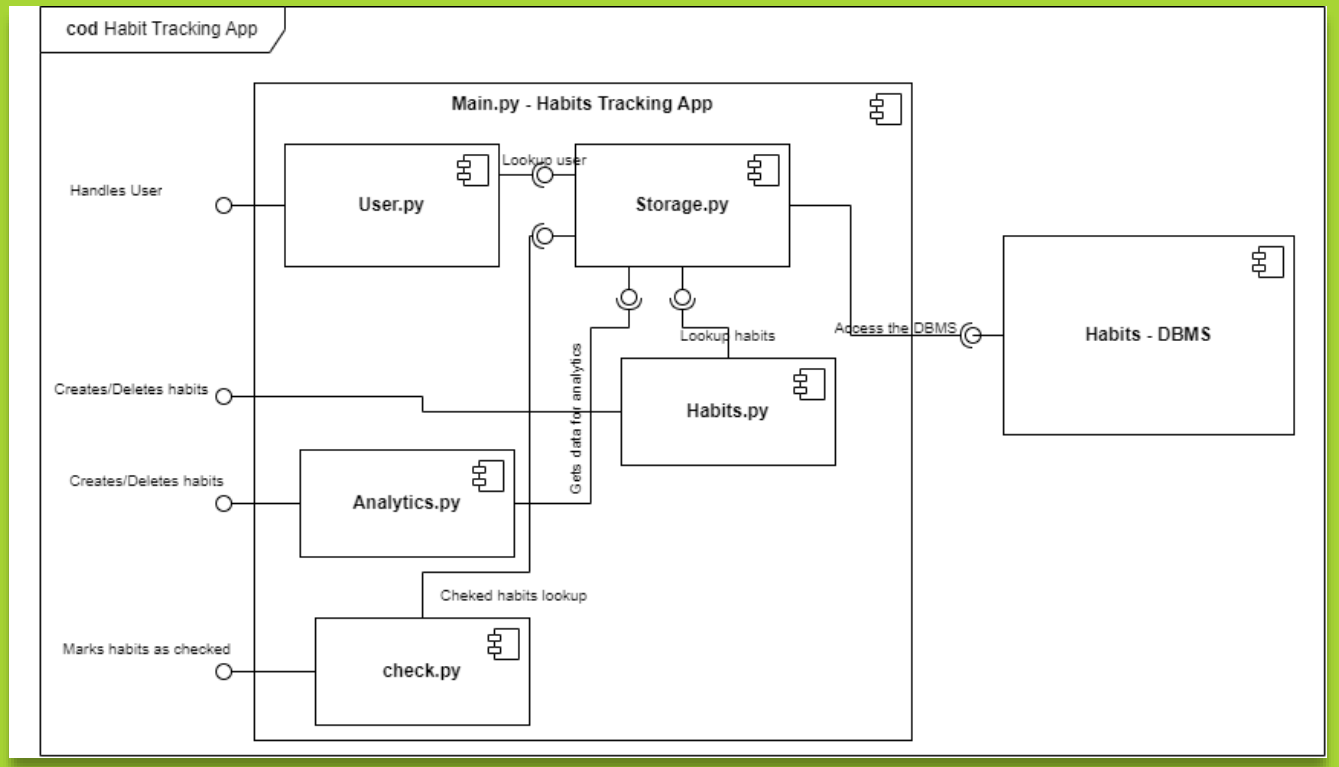
Habit Tracking App – UML Class Diagramm

- Tracking Habit App (THA) is composed by five classes, these are: User, Time, Store, Habits, and Analyses
- User class inherences from the other four classes properties and functions as the central figure of the THA is the user.
- Bulk of the functions are in “store” class, where the connection with a DBMS is open and the communication with it takes place. It imports “psycopg2” module for that purpose.
- “time” class deals with date related calculations. It imports “datatime” module for that purpose, as well as “math” and “random“ modules.
- “Habits“ serves a temporal location for information related to a habit. It does check some properties of input information
- “analytics” class is where the analytics of the habits take place. It imports “pandas” module for that purpose



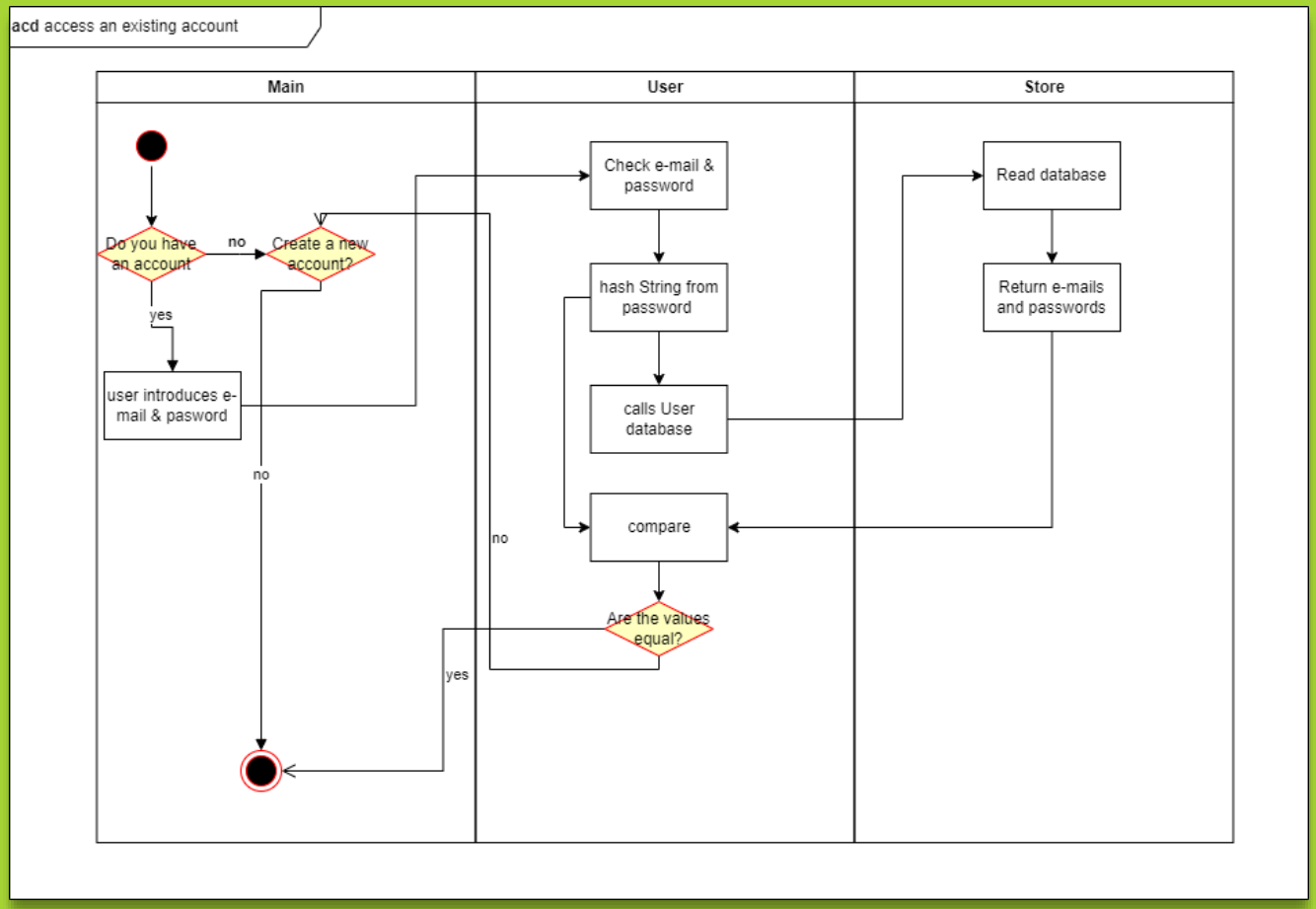
Habit Tracking App – UML Component Diagram

- The file “Main.py” is composed of a set of loops that control the interaction between app and user.
- “Main.py” creates a new object from the class “user”, which inherits the other four classes.
- “storage” class provides the means to access the DBMS for all modules.
- “user” superclass provides the link between “storage” in the “main” environment
- Users have access to some specific functions according of the guidance provided in “Main.py”



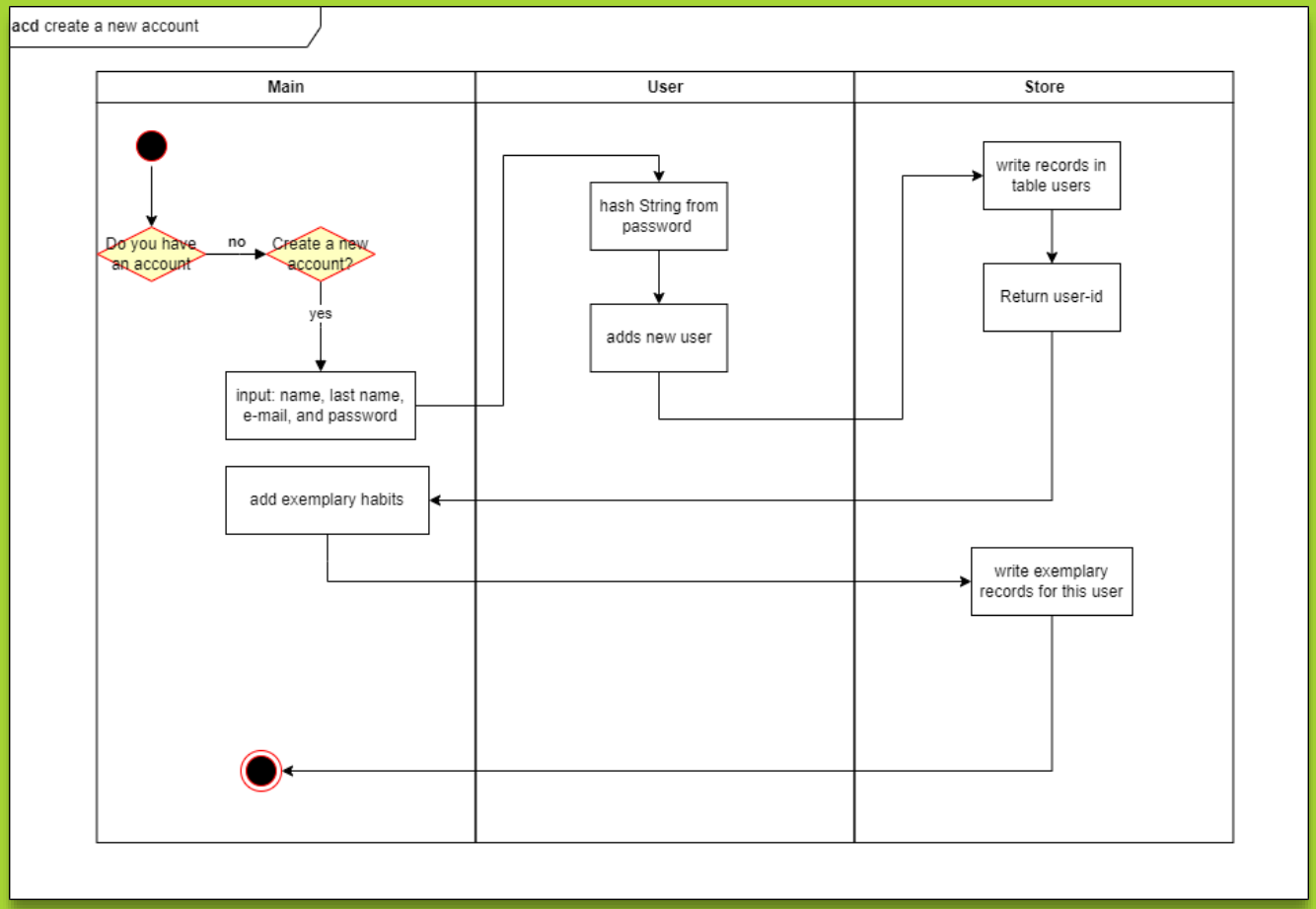
Habit Tracking App – acd “access an existing account”

- UML Activity diagram (acd) to access an existing account.
- For that purpose, three domains are navigated: main, user, store.
- “Main” is where the user is questioned if he has an account.
- The user input password and e-mail; then, in “user” password is converted to hash and compared against the information in the database.
- Information of the database is called in the “store” domain.
- If the values are the same, the user has access to his account.



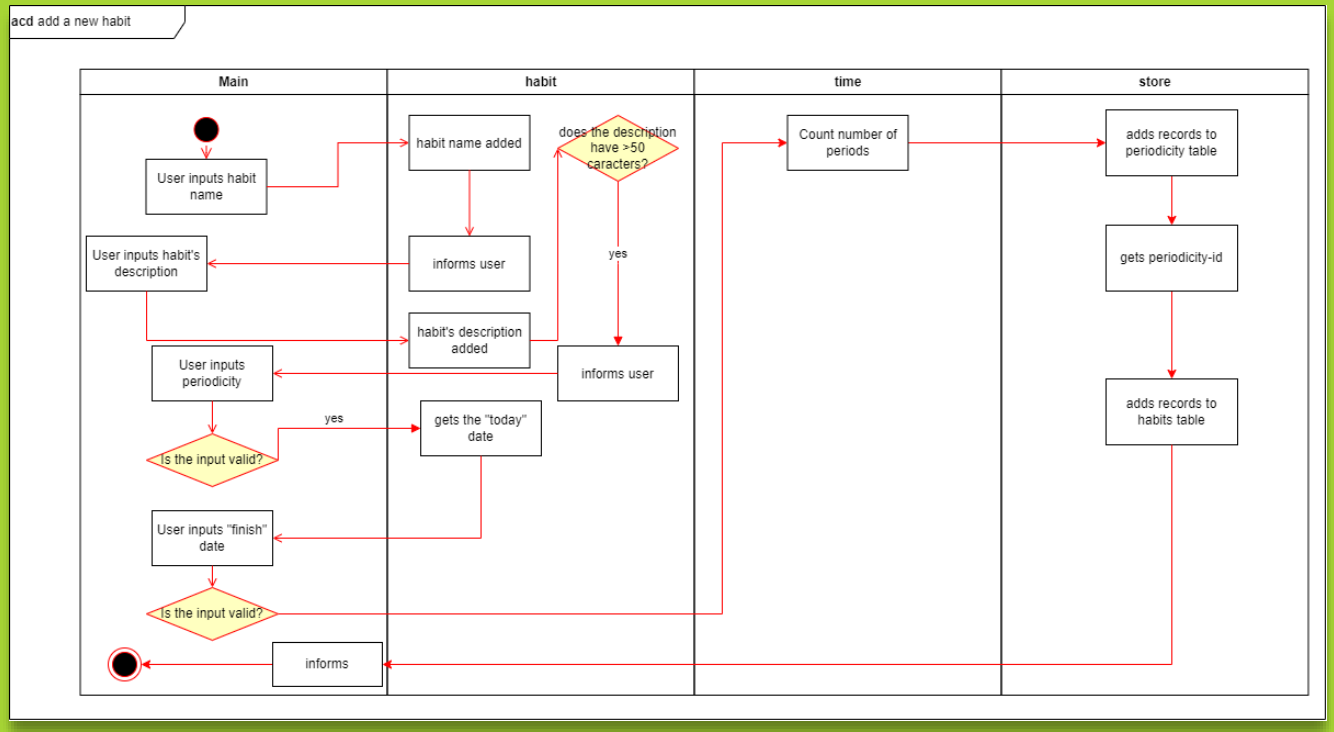
Habit Tracking App – acd “create a new account”

- Three domains are required to create a new account: “Main”, “user”, and “store”.
- Through the control loops the user is requested to answer if he wants to create an account. Likewise, he is required to input name, last name, email, and password.
- These data are passed into “user” where the password is hashed.
- After the information is added to the DBMS in the “store” domain. This returns the “user-id” of the table.
- With the “user-id” a function is called in “main”, such that a set of exemplary data is added for such user.
- When the exemplary records are added in the domain store, then the activity is finished in the domain “Main”.



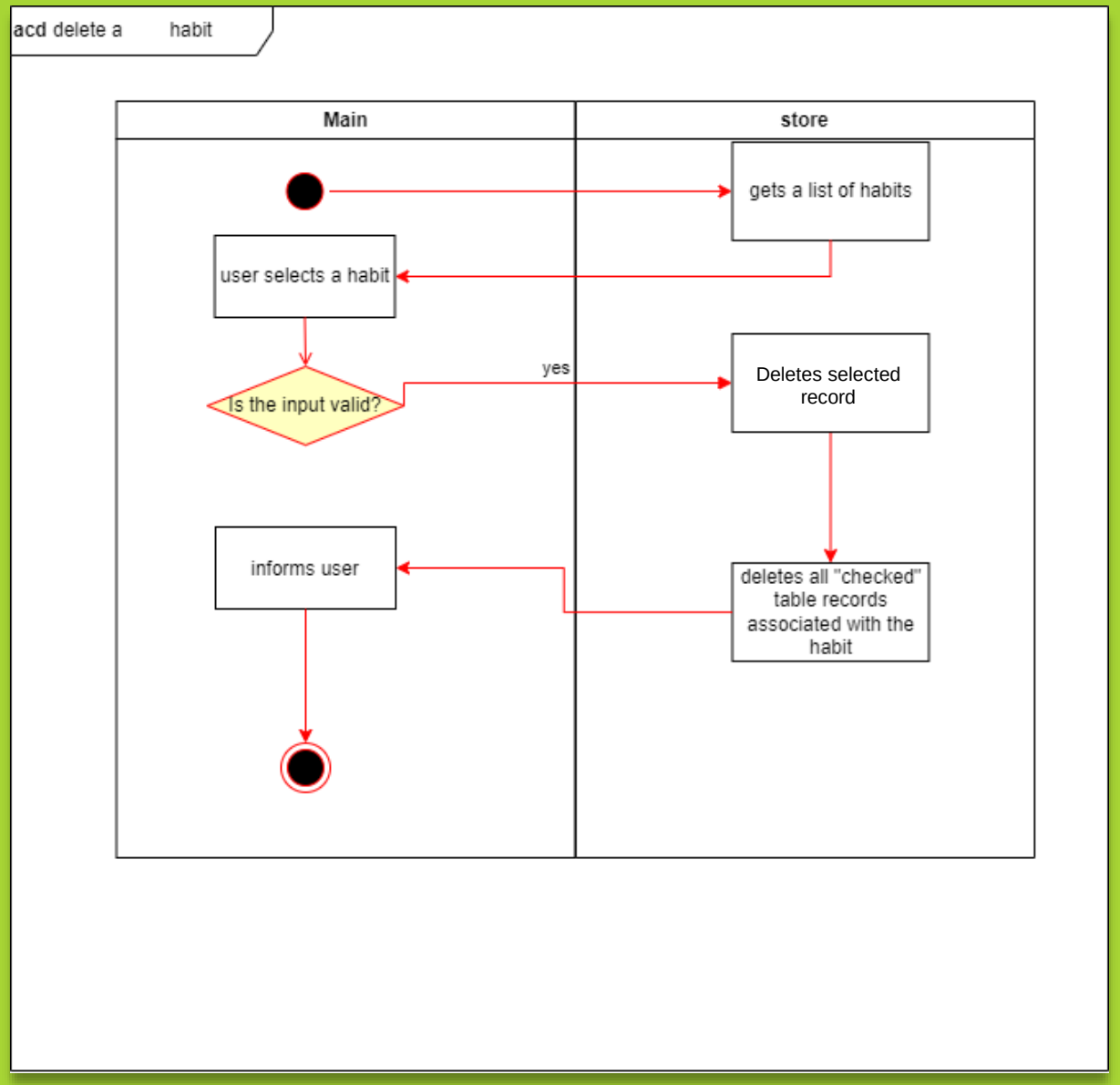
Habit Tracking App – acd “add a new habit”

- Four domains are navigated when creating a new habit: “Main”, “habit”, “time”, and “store”.
- “Main” domain is used to get input from the user and to report results.
- “habit” domain is used to store temporally habit’s name and description. Likewise, it is checked that the description has a minimum of characters.
- “time” domain is used to count the periods *i.e.*, the number of days, weeks, or months depending on the periodicity.
- Domain “store” the information from the database is retrieved.



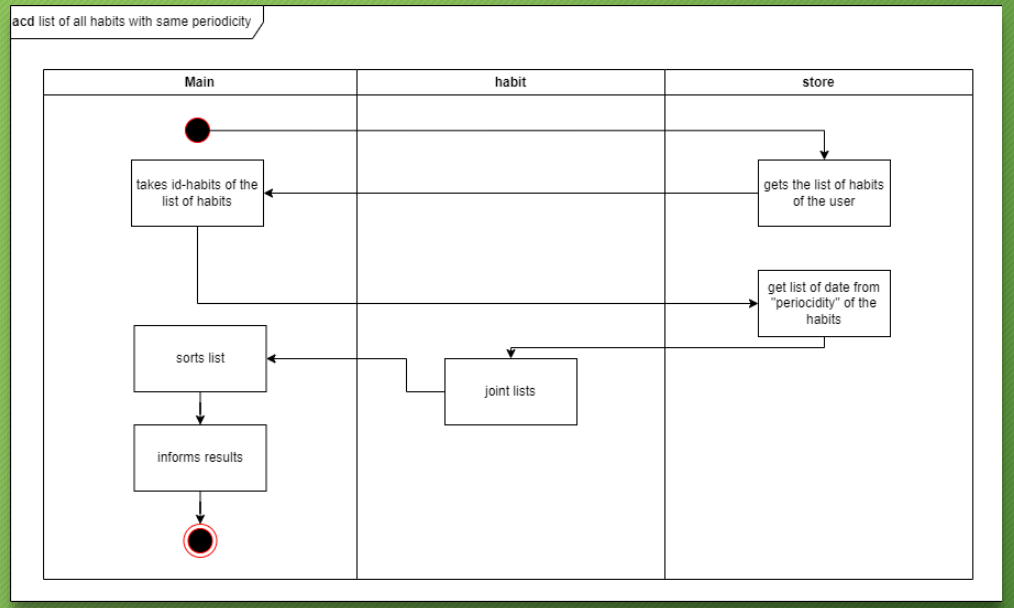
Habit Tracking App – acd “delete a habit”

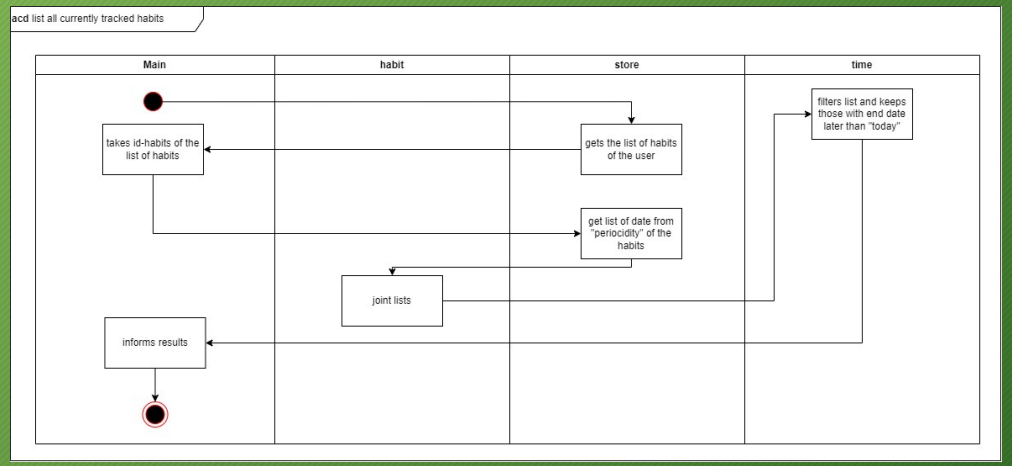
- Two domains are necessary for this activity: “Main” and “store”.
- The process starts in “Main” domain, where is requested immediately in “store” domain the list of habits of the user.
- “store” provides the list and then back in “Main” the user is asked to select an option.
- Still in “Main”, it is checked if the value given is correct.
- Subsequently, it is requested in “store” to delete the selected record in tables “habit” and “checked of the DBMS
- Final step is to inform the user of what was done



Habit Tracking App – acd “list of habit with same periodicity” & “list of current tracked habits”

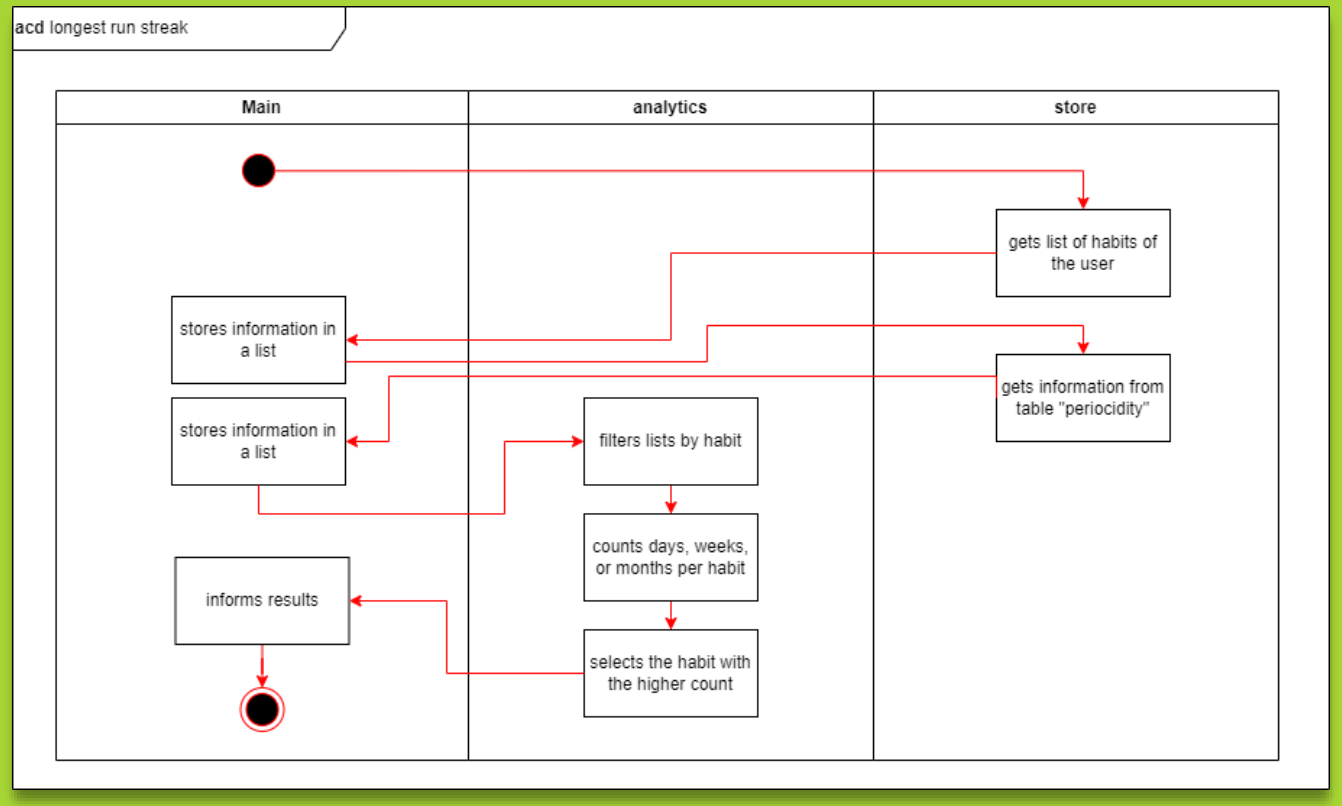
- “time” domain (or check.py) import three modules: “datetime”, “math”, and “random”.
- Module “datetime” is used through the class to convert from string to date datatype and vice versa. It allows to get the “today” date, etc.
- Module “random” is used for the creation of exemplary datasets that the user can use to test the app. It is combined with datetime’s “deltatime” to create the numbers required for the exemplary datasets.
- Module “math” is used to for its function “ceil”





Habit Tracking App – acd “longest run streak”

- Module “analytics” imports “pandas”.
- Pandas is used to for the flexibility provided by its “dataframes” and possibility of grouping records as desired.
- A “dataframe” is created and used to calculate days, weeks, or moths, group them and count continuous days, weeks, or months respectively



Habit Tracking App – acd “longest run streak for a given habit”

- Process starts in “Main” domain and is translated immediately to “store” to get the list of habits of the user.
- From “store” goes back to “Main” and presents the list to user. User selects one habit from the list and is verified that is a valid option.
- As the option is valid, the process goes back to “store” where the list of checked dates of the habit is retrieved from the DBMS.
- After that, the process moves to “analytics” where the consecutive periods are counted. It follows the selection of the longest streak’s habit.
- The process continues in “main” where the results of the activity are presented.